SANDIA REPORT

SAND2009-6125P
Unlimited Release
Printed September 2009

Stable and Efficient Galerkin Reduced Order Models (ROMs) with 'Best Points' Interpolation for Non-Linear Fluid Flow

Irina Kalashnikova, Sandia National Laboratories and Stanford University

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Sandia National Laboratories

Operated for the U.S. Department of Energy by

Sandia Corporation

Albuquerque, NM 87185-0825

Date: 15 September, 2009

To: Distribution

From: Irina Kalashnikova[1,2]

Subject: Stable and Efficient Galerkin Reduced Order Models with "Best" Points Interpolation for Non-Linear Fluid Flow

[1]Aerosciences Department, Sandia National Laboratories, Albuquerque, NM.
[2]Institute for Computational & Mathematical Engineering (iCME), Stanford University, Stanford, CA.
E-mail addresses: ikalash@sandia.gov, irinak@stanford.edu (I. Kalashnikova)

Abstract

The aim of this document is to formulate a stable and efficient reduced order model (ROM) for the *nonlinear*, compressible three-dimensional (3D) Navier-Stokes equations. The work summarized herein is an extension of earlier work [3, 4, 10, 11, 12], initiated under the Sandia National Laboratories' Laboratory Directed Research and Development (LDRD) program to address reduced order modeling for coupled fluid/structure systems. Up to now, only *linearized* fluid equations have been considered. During the months of June - September 2009, the following contributions to the project goal of extending the ROM to non-linear fluid equations were made by the author:

- Application of the "best" points interpolation procedure [15, 16] to a non-linear Galerkin reduced order model (ROM) for fluid flow.
- Implementation and testing of the said "best" points interpolation procedure on a model one-dimensional (1D) convection-diffusion-reaction system of equations for a tubular non-adiabatic reactor [7] in MATLAB, including in particular:
  - Formulation and implementation of an optimization problem to obtain the so-called "hierarchical" and "best" interpolation points.
  - Implementation of the 1D non-linear tubular reactor ROM with interpolation using Fourier cosine and Proper Orthogonal Decomposition (POD) bases.
  - Generation of some numerical results for the non-linear tubular reactor ROM that illustrate the strengths and weaknesses of the "best" points interpolation procedure when different orthogonal bases are employed in the spatial discretization.
- Formulation of a stability-preserving symmetrized ROM for the compressible (*non-linear*) 3D Navier-Stokes equations with appropriate boundary conditions (no-slip and adiabatic wall), including a proof of *ab initio* satisfaction of the second law of thermodynamics, or Clausius-Duhem inequality, for all numerical solutions, a necessary condition for stability.
- Formulation of an efficient interpolation procedure to handle the non-linear terms appearing in the Galerkin-projected, symmetrized 3D compressible Navier-Stokes equations with boundary conditions.

These theoretical and numerical results are presented in detail herein, and may ultimately be incorporated into a journal article.

DOE Funding Statement

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security administration, under Contract DE-AC04-94AL85000.

1 Introduction

In earlier works, namely [3, 4, 10, 11, 12], a Galerkin/Proper Orthogonal Decomposition (POD) Reduced Order Model (ROM) for the three-dimensional (3D) linearized compressible Euler equations was developed, and an extensive mathematical analysis of the ROM with boundary treatment was performed. The focus was on crucial numerical properties of the ROM, namely the well-posedness of the acoustically-reflecting (no-penetration) boundary condition prescribed at the solid wall boundary, and the stability of the Galerkin projection step in the so-called "symmetry inner product". The analysis extended beyond the fluid ROM: stability of the coupled fluid/structure system that arises when one augments the fluid equations with equations for the displacement of a plate over which the fluid is assumed to flow was also considered [10]. These in depth studies of stability and well-posedness led naturally to an analysis of the ROM's convergence. *A priori* error estimates for the computed ROM solution relative to the CFD solution and the exact analytical solution were derived in [11, 12]. To the author's knowledge, these works were the first to address the convergence properties of ROMs derived using the continuous projection approach.

Having formulated and analyzed a linear fluid reduced order model, the goal now is to build a ROM for the non-linear equations of fluid mechanics, namely the 3D compressible Navier-Stokes equations. Several issues must be addressed in the process of developing a reduced order model for these formidable equations:

- The ROM must be stable.
- The ROM must be efficient.

As for their linearized analogs, stability of the Galerkin projection of the non-linear equations can be ensured by defining a transformation that essentially symmetrizes these equations. Following appropriate symmetrization, one can use entropy estimates, namely the Clausius-Duhem entropy inequality, to show *ab initio* satisfaction of the second law of thermodynamics by all numerical solutions to the ROM. This is a necessary condition for stability, which we will term "entropy-stability".

The other issue that must be addressed is efficiency. As discussed in [16] and illustrated herein, for general non-linear partial differential equations (PDEs), the standard Galerkin projection method is no longer efficient in generating reduced order models. This is because the integrals involving the non-linear terms can no longer be precomputed, as in the case for a set of linear equations, but must instead by recomputed at each time or Newton step. The key in circumventing this difficulty is finding a way to handle the Galerkin projection of the non-linear terms without having to recomputed these projections (inner products). To do this, we employ the so-called "best" points interpolation technique [15, 16]. The basic idea is, given a non-polynomial, non-linear function $f(\mathbf{u})$, to represent it efficiently by expanding the function itself linearly in an orthonormal basis.

Having given some motivation for our equations and approach, we now give an outline of what is contained herein. The "best" points interpolation procedure, adapted from [15, 16], is outlined in Section 2 in the context of a one-dimensional (1D) non-linear convection-diffusion-reaction system of PDEs describing non-adiabatic flow through a tubular reactor. We give some results for the tubular reactor ROM with interpolation and a Fourier cosine basis (Section 2.5) and discuss some difficulties encountered when attempting to employ a POD basis on this problem (Section 2.6) In Section 3, we proceed to the 3D compressible Navier-Stokes equations. A change of variables that ensures entropy-stability of the Galerkin projection of the equations *with* appropriate boundary treatment (no-slip and adiabatic wall) is defined using entropy principles [5, 6, 9] (Section 3.4). A procedure to interpolate the non-linear terms that appear in these projected equations is formulated in Sections 3.5–3.6. Conclusions are offered in Section 4. Section 5 is the Appendix, which contains expressions and details not included explicitly in the body of this manuscript.

2 "Best" Points Interpolation Procedure of [15, 16]: Illustration on a 1D Non-Linear Reduced Order Model of a Tubular Reactor [7]

To demonstrate and better understand the properties of the "best" points interpolation [15, 16] prior to formulating its application to the full compressible 3D Navier-Stokes equations, we first illustrate the general interpolation procedure on a simpler one-dimensional (1D) non-linear system of two coupled equations. Given this discussion, it is straightforward to extend the interpolation to each of the non-linear terms in the Navier-Stokes equations (Section 3.6).

2.1 Model 1D Nonlinear Convection-Diffusion-Reaction System of Equations

The problem of interest is a model of a non-adiabatic tubular reactor with a single $A \rightarrow B$ reaction [7]. In dimensionless form, the governing equations, describing the conservation of reactant A and energy for the nonadiabatic tubular reactor with axial mixing, are:

$$\left\{ \begin{array}{ll} \frac{\partial y}{\partial \tau} & = \frac{1}{Pe_m}\frac{\partial^2 y}{\partial s^2} - \frac{\partial y}{\partial s} - Dye^{\gamma - \frac{\gamma}{\theta}}, \qquad s \in (0,1), \tau \in (0,\infty) \\ \frac{\partial \theta}{\partial \tau} & = \frac{1}{Pe_h}\frac{\partial^2 \theta}{\partial s^2} - \frac{\partial \theta}{\partial s} - \beta(\theta - \theta_0) + BDye^{\gamma - \frac{\gamma}{\theta}}, \quad s \in (0,1), \tau \in (0,\infty) \end{array} \right. \tag{1}$$

subject to boundary conditions

$$\left\{ \begin{array}{ll} \frac{\partial y}{\partial s}\big|_{s=0} & = Pe_m(y-1)|_{s=0}, \quad \tau \in (0,\infty) \\ \frac{\partial \theta}{\partial s}\big|_{s=0} & = Pe_h(\theta-1)|_{s=0}, \quad \tau \in (0,\infty) \end{array} \right. \tag{2}$$

$$\left\{ \begin{array}{ll} \frac{\partial y}{\partial s}\big|_{s=1} & = 0, \quad \tau \in (0,\infty) \\ \frac{\partial \theta}{\partial s}\big|_{s=1} & = 0, \quad \tau \in (0,\infty) \end{array} \right. \tag{3}$$

and initial condition

$$y|_{\tau=0} = y_{in}, \qquad \theta|_{\tau=0} = \theta_{in}, \qquad s \in (0,1) \tag{4}$$

Here, y is the dimensionless concentration, θ is the dimensionless temperature, s is the dimensionless axial distance, τ is the dimensionless time, β is the dimensionless heat transfer coefficient, γ is the dimensionless activation energy, D is the Damkohler number, B is the dimensionless heat of reaction, and Pe_m and Pe_h are the Peclet numbers for mass and heat transfer respectively[1]. It is convenient to write (1)–(4) in vector form, as follows:

$$\begin{array}{rll} \frac{\partial \mathbf{u}}{\partial \tau} & = \mathbf{A}^{-1}\frac{\partial^2 \mathbf{u}}{\partial s^2} - \frac{\partial \mathbf{u}}{\partial s} - \mathbf{B}(\mathbf{u}-\mathbf{u}_0) - \mathbf{C}f(\mathbf{u}), & s \in (0,1), \tau \in (0,\infty) \\ \frac{\partial \mathbf{u}}{\partial s}\big|_{s=0} & = \mathbf{A}(\mathbf{u}-\mathbf{1})\big|_{s=0}, & \tau \in (0,\infty) \\ \frac{\partial \mathbf{u}}{\partial s}\big|_{s=1} & = \mathbf{0}, & \tau \in (0,\infty) \\ \mathbf{u} & = \mathbf{u}_{in}, & s \in (0,1) \end{array} \tag{5}$$

where

$$\mathbf{u} \equiv \begin{pmatrix} y \\ \theta \end{pmatrix}, \qquad \mathbf{u}_0 \equiv \begin{pmatrix} y_0 \\ \theta_0 \end{pmatrix} \tag{6}$$

$$\mathbf{A} \equiv \begin{pmatrix} Pe_m & 0 \\ 0 & Pe_h \end{pmatrix}, \qquad \mathbf{B} \equiv \begin{pmatrix} 0 & 0 \\ 0 & \beta \end{pmatrix}, \qquad \mathbf{C} \equiv \begin{pmatrix} D \\ -BD \end{pmatrix}, \qquad \mathbf{1} \equiv \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{7}$$

and

$$f(\mathbf{u}) \equiv ye^{\gamma - \frac{\gamma}{\theta}} \tag{8}$$

Classical numerical techniques [7] illustrate periodic solutions which possess Hopf bifurcations. Figure 1 shows the existence of stable oscillatory solutions as a function of the Damkohler number D when $Pe_m = Pe_h = 5$, $B = 0.50$, $\gamma = 25$, $\beta = 2.5$ and $\theta_0 = 1$. In particular, one can see from this plot that there is a stable orbit bifurcates into a limit cycle at the lower Hopf point, $D = 0.170$.

[1] For more on these parameters, the reader is referred to the "Notation" section of [7].

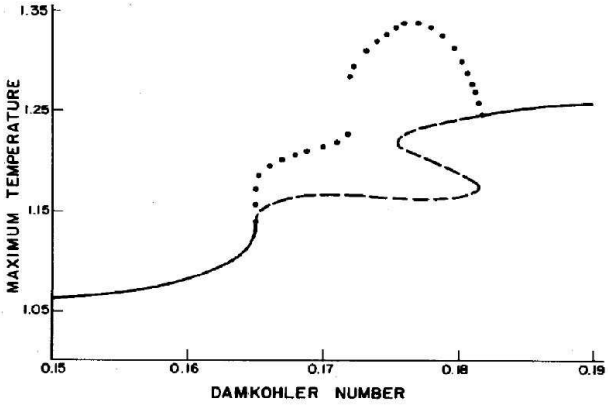


Figure 1: Existence of stable oscillatory solutions to (1) when $Pe_h = Pe_m = 5$, $B = 0.50$, $\gamma = 25$, $\beta = 2.5$, $\theta_0 = 1$

2.2 Weak Formulation and Reduced Order Approximation of (1)

To formulate a reduced order approximation of the solution of (1), begin by expanding $\mathbf{u}$ in an orthonormal vector[2] basis $\{\boldsymbol{\phi}_m^u\}_{m=1}^M \in \mathbb{R}^2$:

$$\mathbf{u}(s,\tau) \approx \mathbf{u}_M(s,\tau) \equiv \sum_{m=1}^{M} a_m(\tau)\boldsymbol{\phi}_m^u(s) \tag{9}$$

where M is the size of the reduced basis. The basis functions $\boldsymbol{\phi}_m^u$ are chosen such that they are orthonormal in some inner product $(\cdot,\cdot)$ (to be specified given the choice of basis), so that $(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \delta_{ij}$, the Dirac delta function.

To obtain the weak form of the equations (1), we project (5) onto the j^{th} mode $\boldsymbol{\phi}_j^u$ in the $(\cdot,\cdot)$ inner product, perform an integration by parts on the diffusion term, and substitute the boundary conditions (2) and (3) into the boundary integral that arises. Doing so gives:

$$\begin{aligned}
0 &= \left(\frac{\partial \mathbf{u}}{\partial \tau} - \mathbf{A}^{-1}\frac{\partial^2 \mathbf{u}}{\partial s^2} + \frac{\partial \mathbf{u}}{\partial s} + \mathbf{B}(\mathbf{u}-\mathbf{u}_0) + \mathbf{C}f(\mathbf{u}), \boldsymbol{\phi}_j^u\right) \\
&= \left(\frac{\partial \mathbf{u}}{\partial \tau}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \boldsymbol{\phi}_j^u}{\partial s}\right) - \langle \mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}\mathbf{n}, \boldsymbol{\phi}_j^u\rangle + \left(\frac{\partial \mathbf{u}}{\partial s}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{B}(\mathbf{u}-\mathbf{u}_0), \boldsymbol{\phi}_j^u\right) + \left(\mathbf{C}f(\mathbf{u}), \boldsymbol{\phi}_j^u\right) \\
&= \left(\frac{\partial \mathbf{u}}{\partial \tau}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \boldsymbol{\phi}_j^u}{\partial s}\right) - \mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}\cdot\boldsymbol{\phi}_j^u\big|_{s=1} + \mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}\cdot\boldsymbol{\phi}_j^u\big|_{s=0} + \left(\frac{\partial \mathbf{u}}{\partial s}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{B}(\mathbf{u}-\mathbf{u}_0), \boldsymbol{\phi}_j^u\right) + \left(\mathbf{C}f(\mathbf{u}), \boldsymbol{\phi}_j^u\right) \\
&= \left(\frac{\partial \mathbf{u}}{\partial \tau}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \boldsymbol{\phi}_j^u}{\partial s}\right) + \mathbf{A}^{-1}\mathbf{A}(\mathbf{u}-\mathbf{1})\big|_{s=0}\cdot\boldsymbol{\phi}_j^u + \left(\frac{\partial \mathbf{u}}{\partial s}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{B}(\mathbf{u}-\mathbf{u}_0), \boldsymbol{\phi}_j^u\right) + \left(\mathbf{C}f(\mathbf{u}), \boldsymbol{\phi}_j^u\right) \\
&= \left(\frac{\partial \mathbf{u}}{\partial \tau}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{A}^{-1}\frac{\partial \mathbf{u}}{\partial s}, \frac{\partial \boldsymbol{\phi}_j^u}{\partial s}\right) + (\mathbf{u}-\mathbf{1})\big|_{s=0}\cdot\boldsymbol{\phi}_j^u + \left(\frac{\partial \mathbf{u}}{\partial s}, \boldsymbol{\phi}_j^u\right) + \left(\mathbf{B}(\mathbf{u}-\mathbf{u}_0), \boldsymbol{\phi}_j^u\right) + \left(\mathbf{C}f(\mathbf{u}), \boldsymbol{\phi}_j^u\right)
\end{aligned} \tag{10}$$

Substituting (9) into (10) and invoking orthonormality of the basis functions $\{\boldsymbol{\phi}_m^u\}_{m=1}^M$ yields the following system of ordinary differential equations (ODEs) for the time-dependent ROM coefficients a_j, $j = 1,\ldots,M$:

$$\begin{aligned}
\dot{a}_j &= -\textstyle\sum_{m=1}^M a_m \mathbf{A}^{-1}\left[\int_0^1 \frac{\partial \boldsymbol{\phi}_m^u}{\partial s}\cdot\frac{\partial \boldsymbol{\phi}_j^u}{\partial s}ds\right] - \sum_{m=1}^M a_m[\boldsymbol{\phi}_m^u(0)\boldsymbol{\phi}_j^u(0)] + \mathbf{1}\cdot\boldsymbol{\phi}_j^u(0) - \sum_{m=1}^M a_m\int_0^1 \frac{\partial \boldsymbol{\phi}_m^u}{\partial s}\cdot\boldsymbol{\phi}_j^u ds \\
&\quad - \textstyle\sum_{m=1}^M a_m \mathbf{B}\left[\int_0^1 \boldsymbol{\phi}_m^u\cdot\boldsymbol{\phi}_j^u ds\right] + \int_0^1 \mathbf{B}\mathbf{u}_0\cdot\boldsymbol{\phi}_j^u ds - \int_0^1 \mathbf{C}f(\mathbf{u}_M)\cdot\boldsymbol{\phi}_j^u ds
\end{aligned} \tag{11}$$

Here, $\dot{a}_j \equiv \frac{da_j}{d\tau}$.

The last term in (11) contains the function $f(\mathbf{u}_M)$, which is non-linear in $\mathbf{u}_M$. In vector form, (11) can be written as:

$$\boxed{\dot{\mathbf{a}}_M = \mathbf{F} - \mathbf{L}\mathbf{a}_M - \mathbf{N}(\mathbf{a}_M)} \tag{12}$$

where

$$\mathbf{a}_M^T \equiv \begin{pmatrix} a_1 & \cdots & a_M \end{pmatrix} \tag{13}$$

[2] Alternatively, one may expand each y and θ in their own scalar, orthonormal bases. This is in fact what is done in Section 2.5, namely (46) and (47). The result is a ROM ODE system involving a total of $2M$ ROM coefficients.

$$F_i = \mathbf{1} \cdot \boldsymbol{\phi}_i^u(0) + \int_0^1 \mathbf{B}\mathbf{u}_0 \cdot \boldsymbol{\phi}_i^u ds \tag{14}$$

$$L_{ij} = \boldsymbol{\phi}_j^u(0)\boldsymbol{\phi}_i^u(0) + \int_0^1 \left[\mathbf{A}^{-1} \frac{\partial \boldsymbol{\phi}_i^u}{\partial s} \cdot \frac{\partial \boldsymbol{\phi}_j^u}{\partial s} + \frac{\partial \boldsymbol{\phi}_j^u}{\partial s} \cdot \boldsymbol{\phi}_i^u + \mathbf{B}\boldsymbol{\phi}_j^u \cdot \boldsymbol{\phi}_i^u \right] ds \tag{15}$$

$$N_i(\mathbf{a}_M) = \int_0^1 \mathbf{C} f\left(\sum_{m=1}^M a_m \boldsymbol{\phi}_m^u \right) \cdot \boldsymbol{\phi}_i^u ds \tag{16}$$

for $i, j = 1, \ldots, M$. In the simpler case when $Pe_h = Pe_m = Pe$, (14) and (15) simplify to:

$$F_i = [\phi_i^u(0)]^1 + [\phi_i^u(0)]^2 + \int_0^1 \beta\theta_0 [\phi_i^u]^2 ds \tag{17}$$

$$\begin{aligned} L_{ij} \quad &= [\phi_i^u(0)]^1 [\phi_j^u(0)]^2 + [\phi_i^u(0)]^2 [\phi_j^u(0)]^2 \\ &+ \int_0^1 \left\{ \tfrac{1}{Pe} \left[\tfrac{\partial [\phi_i^u]^1}{\partial s} \tfrac{\partial [\phi_j^u]^1}{\partial s} + \tfrac{\partial [\phi_i^u]^2}{\partial s} \tfrac{\partial [\phi_j^u]^2}{\partial s} \right] + \left[\tfrac{\partial [\phi_j^u]^1}{\partial s} [\phi_i^u]^1 + \tfrac{\partial [\phi_j^u]^2}{\partial s} [\phi_i^u]^2 \right] + \beta [\phi_j^u]^2 [\phi_i^u]^2 \right\} ds \end{aligned} \tag{18}$$

where $[\phi_i^u]^k$ denotes the k^{th} component of $\boldsymbol{\phi}_i^u$, for $k = 1, 2$. For clarification of the notation in (16), for the function f in (8):

$$f\left(\sum_{m=1}^M a_m \boldsymbol{\phi}_m^u \right) \equiv f\left(\sum_{m=1}^M a_m^y [\phi_m^u]^1, \sum_{m=1}^M a_m^\theta [\phi_m^u]^2 \right) = \left(\sum_{m=1}^M a_m^y [\phi_m^u]^1 \right) \exp\left[\gamma - \gamma \left(\sum_{m=1}^M a_m^\theta [\phi_m^u]^2 \right)^{-1} \right] \tag{19}$$

Since the vector stemming from the non-linear function $f(\mathbf{u})$ (8) depends on $\mathbf{a}_M$, the inner products in (16) *cannot* be pre-computed prior to time-integration of the ROM system (12), as could (and would) be done in the case of linear equations. This greatly reduces the efficiency of this ROM, and motivates one to consider some alternative way to handle the nonlinearity in (16), so that inner products involving $f(\mathbf{u}_M)$ need not be recomputed at each time step.

2.3 Solution for the Interpolation Points: "Best" Points vs. Hierarchical Points

In order to recover efficiency, let us develop the coefficient function approximation for the non-linear terms in (1) by employing the "best" points interpolation of [15, 16]. We outline the general procedure below.

Suppose K snapshots have been taken of the (vector-valued) primal unknown field, at K different times:

$$\mathscr{S}^u \equiv \{ \boldsymbol{\xi}_k^u(s) = \mathbf{u}_h^k(s) : 1 \le k \le K \} \tag{20}$$

Here, the $\mathbf{u}_h^k(s)$ are vectors of state variables at grid point locations, each containing a single solution (snapshot) from the numerical simulation.

Given this set of snapshots of the primal unknown field $\mathbf{u}$, one can construct the following set of snapshots of the non-linear (scalar-valued) function f defined in (8):

$$\mathscr{S}^f \equiv \{ \xi_k^f(s) = f(\mathbf{u}_h^k(s)) : 1 \le k \le K \} \tag{21}$$

We now define the best approximations of the elements in the snapshot set as:

$$f_M^*(\mathbf{u}_h^k(\cdot)) = \arg \min_{w_M \in \text{span}\{\phi_1^f, \ldots, \phi_M^f\}} ||f(\mathbf{u}_h^k(\cdot)) - w_M||, \qquad 1 \le k \le K \tag{22}$$

where $\{\phi_m^f\}_{m=1}^M$ is an orthonormal (in this case, scalar) basis for f. Orthonormality of the ϕ_m^f implies that

$$f_M^*(\mathbf{u}_h^k(x)) = \sum_{m=1}^M \alpha_m^k \phi_m^f(s), \qquad 1 \le k \le K \tag{23}$$

where

$$\alpha_m^k = (\phi_m^f, f(\mathbf{u}_h^k(\cdot))), \qquad m = 1,...,M, 1 \le k \le K \tag{24}$$

The "best" interpolation points [15, 16] $\{s_m^{bp}\}_{m=1}^M$ are defined as the solution to the following optimization problem:

$$\begin{array}{c} \min_{s_1^{bp},...,s_M^{bp} \in \Omega} \sum_{k=1}^K \left|\left| f_M^*(\mathbf{u}_h^k(\cdot)) - \sum_{m=1}^M \beta_m^k(s_1^{bp},...,s_M^{bp})\phi_m^f \right|\right|^2 \\ \sum_{n=1}^M \phi_n^f(s_m^{bp})\beta_n^k(s_1^{bp},...,s_M^{bp}) = f(\mathbf{u}_h^k(s_m^{bp})), \qquad 1 \le m \le M, 1 \le k \le K \end{array} \tag{25}$$

Substituting (23) into (25) and invoking the orthonormality of the $\{\phi_m^f\}_{m=1}^M$, we obtain:

$$\boxed{\begin{array}{c} \min_{s_1^{bp},...,s_M^{bp} \in \Omega} \sum_{k=1}^K \sum_{m=1}^M (\alpha_m^k - \beta_m^k(s_1^{bp},...,s_M^{bp}))^2 \\ \sum_{n=1}^M \phi_n^f(s_m^{bp})\beta_n^k(s_1^{bp},...,s_M^{bp}) = f(\mathbf{u}_h^k(s_m^{bp})), \qquad 1 \le m \le M, 1 \le k \le K \end{array}} \tag{26}$$

i.e., the set of points $\{s_m^{bp}\}_{m=1}^M$ is determined to minimize the average error between the interpolants $f_M(\cdot)$ and the best approximations $f_M^*(\cdot)$. For implementational purposes, it is useful to rewrite (26) as

$$\begin{array}{c} \min_{s_1^{bp},...,s_M^{bp} \in \Omega} \sum_{q=1}^Q (\tilde{\alpha}_q - \tilde{\beta}_q(s_1^{bp},...,s_M^{bp}))^2 \\ \sum_{n=1}^M \phi_n^f(s_m^{bp})\tilde{\beta}_{(k-1)M+m}(s_1^{bp},...,s_M^{bp}) = f(\mathbf{u}_h^k(s_m^{bp})), \qquad 1 \le m \le M, 1 \le k \le K \end{array} \tag{27}$$

where $Q = MK$ and, for $1 \le m \le M$, $1 \le k \le K$,

$$\begin{array}{l} \alpha_m^k = \tilde{\alpha}_{(k-1)M+m} \\ \beta_m^k = \tilde{\beta}_{(k-1)M+m} \end{array} \tag{28}$$

The solution to the least-square optimization problem (27) can be found using the Levenberg-Marquardt (LM) algorithm[3]. According to [16], the optimal solution is typically reached in less than fifteen iterations of the LM algorithm.

As noted in [15], the solutions to (27) are in general non-unique, as the objective function defining the "best" points is usually non-convex. As a consequence, any iterative minimization algorithm used to solve (27) is very sensitive to the initial guess.

One systematic approach that works well for selecting the initial guess for (27) is to first compute the so-called "hierarchical" interpolation points[4], $\{s_m^{hp}\}_{m=1}^M$ and then use these as the initial guess in finding the "best" points. The hierarchical points are less expensive to construct than the "best" points $\{s_m^{bp}\}_{m=1}^M$, as they are computed one at a time by solving a sequence of univariate optimization problems. They also exhibit the nice property that $\{s_1^{hp},...,s_m^{hp}\} \subset \{s_1^{hp},...,s_{m+1}^{hp}\}$ for $m = 1,...,M-1$.

The solution procedure for the hierarchical points is as follows. To obtain the first hierarchical point, s_1^{hp} one computes the minimizer of the following (univariate, or 1D) optimization problem:

$$\begin{array}{c} \min_{s_1^{hp} \in \Omega} \sum_{k=1}^K (\alpha_1^k - \beta_1^k(s_1^{hp}))^2 \\ \phi_1^f(s_1^{hp})\beta_1^k(s_1^{hp}) = f(\mathbf{u}_h^k(s_1^{hp})), \qquad 1 \le k \le K \end{array} \tag{29}$$

Then, for $L = 2,...,M$, one finds s_L^{hp} and appends it to the sequence $\{s_1^{hp},...,s_{L-1}^{hp}\}$ already computed, where s_L^{hp} is defined as the minimizer of

$$\begin{array}{c} \min_{s_L^{hp} \in \Omega} \sum_{k=1}^K \sum_{l=1}^L (\alpha_l^k - \beta_l^k(s_L^{hp}))^2 \\ \sum_{l=1}^L \phi_l^f(s_m^{hp})\beta_l^k(s_m^{hp}) = f(\mathbf{u}_h^k(s_m^{hp})), \qquad 1 \le m \le L-1, 1 \le k \le K \\ \sum_{l=1}^L \phi_l^f(s_L^{hp})\beta_l^k(s_L^{hp}) = f(\mathbf{u}_h^k(s_L^{hp})), \qquad 1 \le k \le K \end{array} \tag{30}$$

[3] E.g., by employing the `lsqnonlin` function in MATLAB's optimization toolbox.

[4] For more on the hierarchical points, the reader is referred to Section 2.2.3 of [15].

where $L = 2, ..., M$ and α_l^k is as defined in (24).

As will be illustrated in Section 2.5, the objective functions in (30) are, like the objective function in (27), in general non-convex, meaning they possess multiple local minima. However, as each minimization (30) is univariate, one could, rather than using an iterative optimization procedure to obtain local minima, compute the global minimizers that solve (30). This idea is explored further in Section 2.5.

2.4 Reduced Order Approximation to (1) with Interpolation

Given the "best" points for f, i.e., the solutions to (27) (or any set of interpolation points), call them $\{s_m^f\}_{m=1}^M$, it is straight forward to apply the interpolation outlined in Section 2.3 to the non-linear function $f(\mathbf{u})$ (8). We begin by computing snapshots for the non-linear function f in (8). From these snapshots we compute the interpolation points $\{s_m^f\}_{m=1}^M$ following the approach outlined in Section 2.3 (see also Section 2 of [16]). Given $\{s_m^f\}_{m=1}^M$ and $\{\phi_m^f\}_{m=1}^M$, one obtains the so-called "cardinal functions" $\{\psi_m^f\}_{m=1}^M$ by solving the following linear system

$$\boldsymbol{\phi}_M^f(s) = \mathbf{A}\boldsymbol{\psi}_M^f(s) \tag{31}$$

where $\boldsymbol{\phi}_M^f(s) = (\phi_1^f(s), ..., \phi_M^f(s))^T$ and $\boldsymbol{\psi}_M^f(s) = (\psi_1^f(s), ..., \psi_M^f(s))^T$, and $A_{ij} = \phi_j^f(z_i)$. Note that (31) is well-defined, as the basis for f, like f itself, is *scalar*. Note also that the cardinal functions satisfy $\psi_j(z_i) = \delta_{ij}$.

Given the interpolation points $\{s_m^f\}$ and the cardinal functions $\{\psi_m^f\}$, one can approximate f as:

$$f(\mathbf{u}) \approx f_M(\mathbf{u}) = \sum_{m=1}^{M} f(\mathbf{u}(s_m^f))\psi_m^f \in \mathbb{R} \tag{32}$$

so that

$$f_M = \sum_{m=1}^{M} f\left(\sum_{n=1}^{M} a_n(t)\boldsymbol{\phi}_n^u(s_m^f)\right)\psi_m^f \tag{33}$$

The projection of f_M onto the l^{th} POD mode for $\mathbf{u}$ can be written in matrix/vector form. To do this, note that, for a general function $f_M(\mathbf{a}_M)$ and for $l = 1, ..., M$, we have that:

$$\begin{aligned}(\boldsymbol{\phi}_l^u, \mathbf{C}f_M(\mathbf{a}_N)) &= \left(\boldsymbol{\phi}_l^u, \textstyle\sum_{m=1}^M \mathbf{C}f\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n^u(s_m^f)\right)\psi_m^f\right) \\ &= \textstyle\sum_{m=1}^M \left[\int_\Omega \boldsymbol{\phi}_l^u \cdot \mathbf{C}\psi_m^f d\Omega\right] f\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n^u(s_m^f)\right)\end{aligned} \tag{34}$$

Remark that (34) is a matrix/vector product of the form $\mathbf{G}f\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n^u(s_m^f)\right)$ where

$$G_{nm} = \int_\Omega \boldsymbol{\phi}_n^u \cdot \mathbf{C}\psi_m^f d\Omega \tag{35}$$

for $1 \le m, n \le M$ (so that $\mathbf{G} \in \mathbb{R}^{M\times M}$).

It follows that, with the interpolation procedure employed here, our ODE system for the ROM coefficients is not (12) but rather

$$\boxed{\dot{\mathbf{a}}_M = \mathbf{F} - \mathbf{L}\mathbf{a}_M - \mathbf{G}f(\mathbf{D}^f\mathbf{a}_M)} \tag{36}$$

where $\mathbf{F}$ and $\mathbf{L}$ are defined in (14) and (15) respectively, the entries of $\mathbf{G}$ are given by (35), and

$$\mathbf{D}^f \equiv \begin{pmatrix} \boldsymbol{\phi}_1^u(s_1^f) & \dots & \boldsymbol{\phi}_M^u(s_1^f) \\ \vdots & \ddots & \vdots \\ \boldsymbol{\phi}_1^u(s_M^f) & \dots & \boldsymbol{\phi}_M^u(s_M^f) \end{pmatrix} \in \mathbb{R}^{2M\times M} \tag{37}$$

To clarify the notation in (36), namely what is meant by a function f of a vector:

$$f(\mathbf{D}^f \mathbf{a}_M) \equiv f\begin{pmatrix} \sum_{m=1}^M \boldsymbol{\phi}_m^u(s_1^f)a_m \\ \vdots \\ \sum_{m=1}^M \boldsymbol{\phi}_m^u(s_M^f)a_m \end{pmatrix} \equiv \begin{pmatrix} f\left(\sum_{m=1}^M \boldsymbol{\phi}_m^u(s_1^f)a_m\right) \\ \vdots \\ f\left(\sum_{m=1}^M \boldsymbol{\phi}_m^u(s_M^f)a_m\right) \end{pmatrix} \in \mathbb{R}^M \tag{38}$$

where $f\left(\sum_{m=1}^M \boldsymbol{\phi}_m^u(s_i^f)a_m\right)$ is defined as in (19).

Essentially, in the interpolation procedure outlined here, recomputation of inner products (projection) of the nonlinear terms at each time (or Newton) step is replaced by evaluation of the basis functions at the (pre-computed) interpolation points (37). There is also a matrix inversion (31) involved in solving for the cardinal functions $\{\psi_m^f\}_{m=1}^M$ (31). This is the key difference between the ROM with interpolation (36) and the ROM without interpolation (12), and what makes (36) far more efficient. The formulation and solution of the ROM with interpolation, including computation of the "best" points and time-integration, is summarized in Algorithm 1 below.

Algorithm 1 Summary of ROM solution procedure of (1) with "best" points interpolation

1. Compute a set of K snapshots for the primal unknown field $\mathbf{u}$:

$$\mathscr{S}^u \equiv \{\boldsymbol{\xi}_k^u(s) = \mathbf{u}_h^k(s) : 1 \le k \le K\} \tag{39}$$

2. Given this set of snapshots of the primal unknown field $\mathbf{u}$, compute the following set of snapshots of the non-linear function f (8) from (39):

$$\mathscr{S}^f \equiv \{\xi_k^f(s) = f(\mathbf{u}_h^k(x)) : 1 \le k \le K\} \tag{40}$$

3. Compute an orthonormal basis $\{\phi_1^f, ..., \phi_M^f\}$ for f.
4. For the nonlinear function $f(\mathbf{u})$ in (8), apply (for instance) the Levenberg-Marquandt (LM) algorithm to solve (27) for the "best" points $\mathbf{s}^{bp} = \{s_1^{bp}, ..., s_M^{bp}\}$.
5. Compute an orthonormal basis $\{\phi_1^u, ..., \phi_M^u\}$ for $\mathbf{u}$.
6. Compute the matrix $\mathbf{L}$ and vector $\mathbf{F}$ from (15) and (16) respectively.
7. Compute $\mathbf{A} \equiv \mathbf{A}(\mathbf{s}^{bp})$ at the best points, with:

$$A_{mn} = \phi_n^f(s_m^f) \tag{41}$$

8. Compute the set of cardinal functions $\{\psi_1^f, ..., \psi_M^f\}$ by solving $\boldsymbol{\phi}_M^f = \mathbf{A}\boldsymbol{\psi}_M^f$.
9. Compute $\mathbf{G}$ from

$$G_{nm} = \int_\Omega \boldsymbol{\phi}_n^u \cdot \mathbf{C}\psi_m^f d\Omega \tag{42}$$

10. Compute $\mathbf{D}^f$ from

$$D_{mn}^f = \phi_n^u(z_m^f) \tag{43}$$

11. Advance the following ODE system forward in time using a standard time-integration scheme (e.g., Euler, Runge-Kutta, etc.).

$$\dot{\mathbf{a}}_N + \mathbf{L}\mathbf{a}_N - \mathbf{F} + \mathbf{G}f(\mathbf{D}^f \mathbf{a}_N) = \mathbf{0} \tag{44}$$

(Note that Newton's method is not required if an explicit scheme is employed.)

2.5 Some Numerical Results for (8) with a Fourier Cosine Reduced Basis

Since we are interested in formulating a *Galerkin* Reduced Order Model (ROM), we seek a basis that satisfies the boundary conditions, namely (2) and (3). It turns out that a Fourier cosine basis, defined by

$$\phi_m(s) = \cos(\pi(m-1)s), \qquad m = 1, ..., M \tag{45}$$

for $0 \leq s \leq 1$ satisfies these boundary conditions. To generate some preliminary numerical results, we will therefore employ a scalar Fourier cosine basis for each y and θ:

$$y(\tau,s) \approx y_M(\tau,s) = \sum_{m=1}^{M} a_m^y(\tau)\cos(\pi(m-1)s) \tag{46}$$

$$\theta(\tau,s) \approx \theta_M(\tau,s) = \sum_{m=1}^{M} a_m^\theta(\tau)\cos(\pi(m-1)s) \tag{47}$$

It is well-known that the Fourier cosine basis is orthonormal in the L_2 inner product; hence, we will take as $(\cdot,\cdot)$ the $L_2([0,1])$ inner product. It follows that the coefficients in (46) and (47) are given by

$$a_m^y(\tau) = (y(\tau,s),\cos(\pi(m-1)s)_{L_2([0,1])} \equiv \int_0^1 y(\tau,s)\cos(\pi(m-1)s)ds \tag{48}$$

and

$$a_m^\theta(\tau) = (\theta(\tau,s),\cos(\pi(m-1)s)_{L_2([0,1])} \equiv \int_0^1 \theta(\tau,s)\cos(\pi(m-1)s)ds \tag{49}$$

respectively.

For concreteness and to generate some numerical results, we fix the properties in the equations (1) to those summarized in Table 1. As one can infer from Figure 1, the solution exhibits a limit cycle for $D = 0.170$. One of our goals is to see

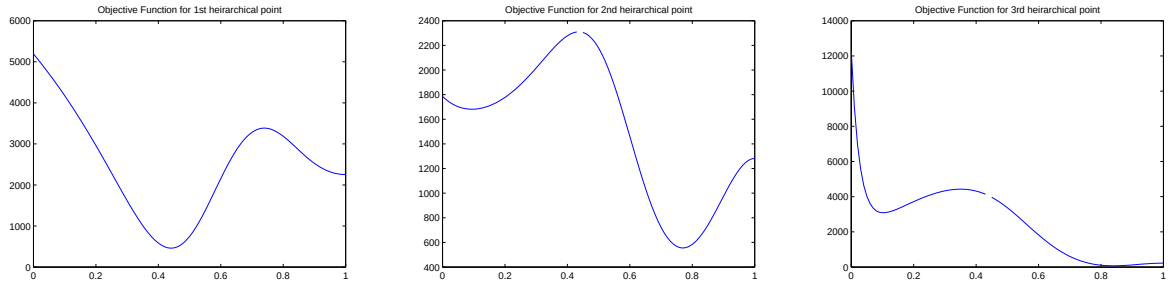
Table 1: Fluid properties used in the numerical solution of (1)

Property	Symbol	Value
Peclet number for heat transfer	Pe_h	5.00
Peclet number for mass transfer	Pe_m	5.00
Dimensionless heat of reaction	B	0.50
Dimensionless activation energy	γ	25.0
Dimensionless heat transfer coefficient	β	2.50
Reference dimensionless temperature	θ_0	1.00
Damkohler number	D	0.17

if the ROM solution with interpolation captures this limit cycle correctly.

The reduced order model for which we give numerical results was generated by taking $K = 701$ snapshots of the solution fields y and θ, at time increments $d\tau = 0.25$ apart. Numerical tests reveal that, to capture the correct limit cycle, a reduced basis Fourier cosine basis of size at least $M = 6$ modes is required. Below, we give some results for $M = 6$ and $M = 10$.

Tables 2 and 3 give the uniform, hierarchical and "best" points for a basis of size $M = 6$ and $M = 10$ respectively, and the nonlinear function (8). The "best" points are computed by solving the optimization problem (27) using the `lsqnonlin` function in MATLAB's optimization toolbox, and with the hierarchical points as the initial guess. We note that the objective function in (27) is not necessarily convex, so it may possess multiple local minima, which implies that the "best" points are non-unique. The same is true for the objective function that defines the hierarchical points. Since in this latter case of the hierarchical points, the minimization is univariate, it is possible to obtain hierarchical points that are global minima of the relevant objective function (third column of Tables 2 and 3). These points can be quite different, as one can see by comparing columns two and three of Tables 2 and 3. Indeed, the following figure (Figure 2) shows that the objective function defining the second and third hierarchical points (and subsequent hierarchical points) possesses multiple local minima. An interesting, perhaps somewhat surprising, observation is that it turns out not to matter which hierarchical points, the local minimizers (column two of Tables 2 and 3) or the global minimizers (column three of Tables 2 and 3) are used as the initial guess to obtain the "best" points in column four of these tables. Selecting s^{unif} (the uniform points) as the initial guess produces in general a different set of "best" points, however.



(a) s_1^{hp} (b) s_2^{hp} (c) s_3^{hp}

Figure 2: Objective functions defining the first three hierarchical points s_1^{hp}, s_2^{hp} and s_3^{hp}

The ROM ODE system resulting from the discretization (36) was advanced forward in time using a nonlinear fourth order Runge-Kutta (RK-4) time integration scheme, with time step $\Delta\tau = 10^{-3}$.

Figures 3 and 5 show the $L_2([0,1])$ errors in the ROM solutions computed using $M = 6$ and $M = 10$ Fourier cosine modes respectively with the interpolation outlined in Section 2.3. Errors are computed relative to the snapshots for uniformly spaced interpolation points versus the "best" points. It is found that using the "best" points reduces the error by an order of magnitude in general. One can also see by comparing Figures 3 and 5 that there is greater payoff in using the "best" points for smaller M.

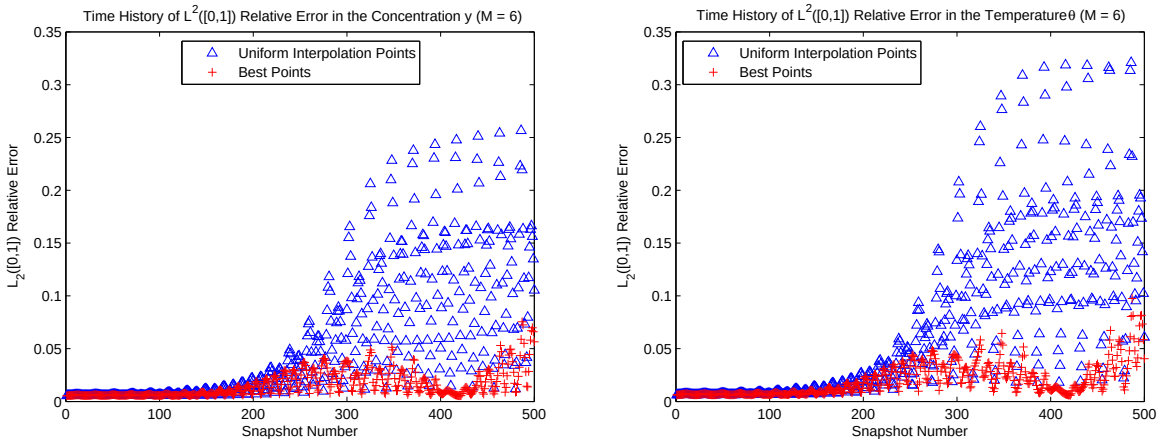
Figures 4 and 6 depict the limit cycle computed by the 6 and 10 mode ROMs (respectively) with "best" points interpolation. There is a slight phase error when $M = 6$ because so few modes are employed; nonetheless, it is clear that the nonlinear behavior, namely the limit cycle, is captured. There is excellent agreement between the ROM limit cycle and the snapshot limit cycle for $M = 10$ (Figure 6): the non-linear behavior is captured with the correct phase/magnitude.

Table 2: Uniform (s^{unif}), hierarchical (s^{hp}) and "best" (s^{bp}) points for $M = 6$

s^{unif}	s^{hp} (local minimizer)	s^{hp} (global minimizer)	s^{bp}
0.0000	0.0291	0.2000	0.0572
0.2000	0.1202	0.2500	0.2465
0.4000	0.2211	0.4400	0.4203
0.6000	0.3252	0.7200	0.5909
0.8000	0.4311	0.7700	0.7578
1.0000	0.5378	0.7800	0.9211

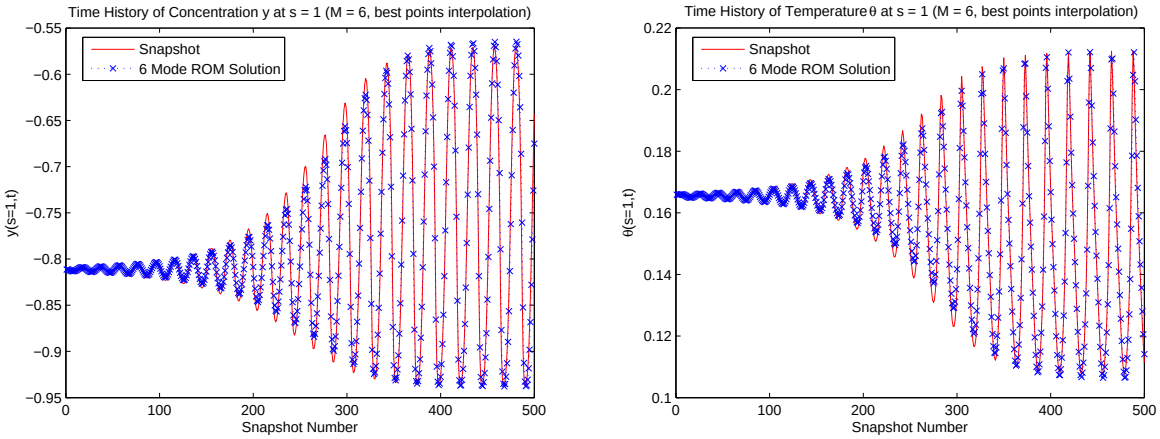
Table 3: Uniform (s^{unif}), hierarchical (s^{hp}) and "best" (s^{bp}) points for $M = 10$

s^{unif}	s^{hp} (local minimizer)	s^{hp} (global minimizer)	s^{bp}
0.0000	0.0945	0.0072	0.0291
0.1111	0.1021	0.0400	0.1202
0.2222	0.2568	0.2000	0.2211
0.3333	0.3355	0.2500	0.3252
0.4444	0.4407	0.4200	0.4311
0.5556	0.5490	0.4400	0.5378
0.6667	0.6659	0.5000	0.6443
0.7778	0.7014	0.7700	0.7504
0.8889	0.8529	0.7800	0.8560
1.0000	0.9157	0.8700	0.9564



(a) Concentration y (b) Temperature θ

Figure 3: Time history of $L^2([0,1])$ relative errors in 6 mode ROM solution with interpolation using uniform points vs. "best" points



(a) Concentration y (b) Temperature θ

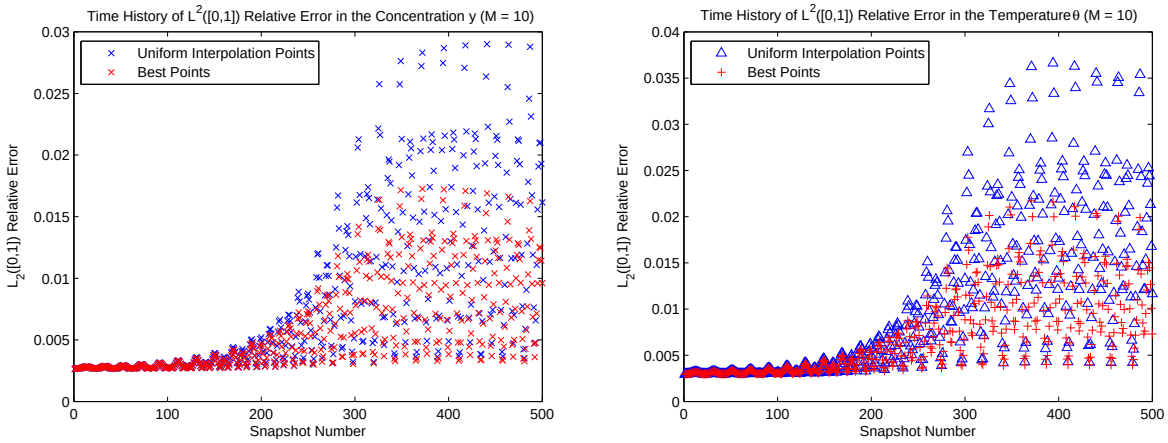
Figure 4: Correct computation of limit cycles by a 6 mode ROM with "best" points interpolation

2.6 Other Orthogonal Bases: Some Difficulties with a Proper Orthogonal Decomposition (POD) Basis for the Tubular Reactor ROM

We end the discussion of the 1D non-linear tubular rector ROM with some comments regarding the performance of the ROM with a Proper Orthogonal Decomposition (POD) basis, instead of a spectral basis like the Fourier cosine basis (45). Let us first give a brief overview of POD.

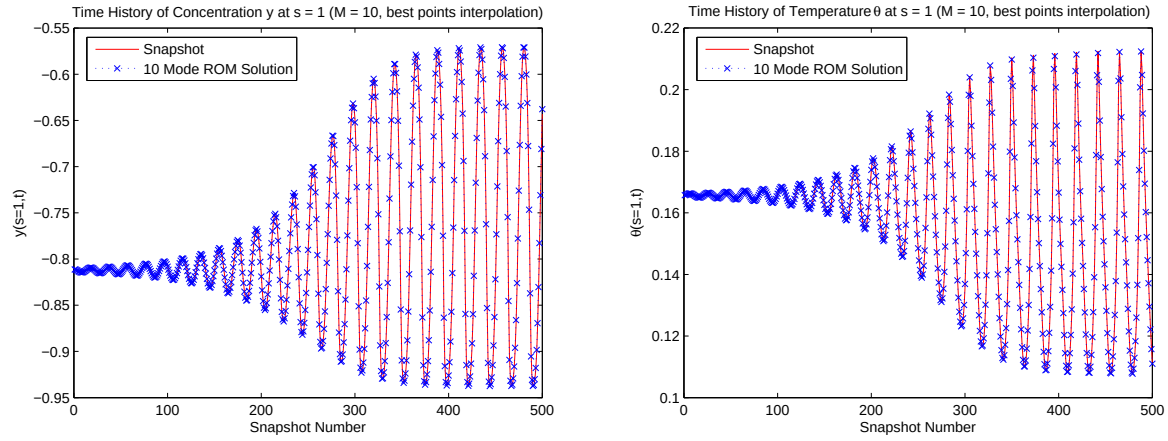
Discussed in detail in Lumley [14] and Holmes *et. al.* [8], POD is a mathematical procedure that, given an ensemble of data, constructs a basis for that ensemble that is optimal in a well-defined sense. A POD basis of order $M << N$ is a set of functions $\{\boldsymbol{\phi}_i : i = 1,2,\ldots,M\}$ that is the "best" linear basis for describing the original ensemble. Mathematically, POD seeks an M-dimensional ($M << N$) subspace spanned by the set $\{\boldsymbol{\phi}_i\}$ such that the projection of the difference between the ensemble of snapshots (realizations) of the flow field $\mathbf{u}^k$ and its projection onto the subspace is minimized on average. It is a well-known result [8, 12] that the solution to this minimization problem reduces to the eigenvalue problem

$$\mathscr{R}\boldsymbol{\phi} = \lambda\boldsymbol{\phi} \tag{50}$$



(a) Concentration y (b) Temperature θ

Figure 5: Time history of $L^2([0,1])$ relative errors in 10 mode ROM solution with interpolation using uniform points vs. "best" points



(a) Concentration y (b) Temperature θ

Figure 6: Correct computation of limit cycles by a 10 mode ROM with "best" points interpolation

where

$$\mathscr{R}\boldsymbol{\phi} = \langle \mathbf{u}^k(\mathbf{u}^k, \boldsymbol{\phi}) \rangle \tag{51}$$

and $\langle \cdot \rangle$ denotes a time-averaging operator. The operator $\mathscr{R}$ is self-adjoint and non-negative definite. If one further assumes that $\mathscr{R}$ is compact, then there exists a countable set of non-negative eigenvalues λ_i with associated eigenfunctions $\boldsymbol{\phi}_i$, orthonormal in the relevant inner product. In building a ROM, one is interested in truncating the POD basis and retaining only the $M << N$ most energetic modes. It can be shown [8, 14] that the set of M eigenfunctions, or POD modes, $\{\boldsymbol{\phi}_i : i = 1, 2, \ldots, M\}$ is optimal in the sense that it describes more energy (on average) of the ensemble than any other linear basis of the same dimension M. The compression of the ensemble energy into a minimum number of modes is what makes the POD basis attractive for reduced order modeling. We note the POD basis $\{\boldsymbol{\phi}_i : i = 1, 2, \ldots, M\}$ just described is *not* complete. It is, however, complete in an average sense, that is $\left\langle \left|\left| \mathbf{u}^k - \sum_j (\mathbf{u}^k, \boldsymbol{\phi}_j)\boldsymbol{\phi}_j \right|\right| \right\rangle = 0$ for $M = N$.

Numerical experiments with a POD basis suggest that POD is a poor choice of ROM basis for this problem. In particular, it is observed that:

- The method of snapshots appears to break down when computing bases of M greater than approximately 30 for the $K = 700$ snapshots of the field $\mathbf{u}$. Non-orthogonal modes begin to appear.
- The Fourier basis seems to be much more efficient at representing the nonlinear function $f(\mathbf{u})$ (8) than the POD basis.

The second point is likely the primary cause of the trouble. Indeed, POD is optimal in representing $\mathbf{u}$ but not necessarily $f(\mathbf{u})$ so its inadequacy is not entirely surprising. The excellent results with a Fourier basis can likely be attributed to the smoothness of this basis, as well as the fact that the exact derivatives of the basis functions are available.

We emphasize that the inadequacy of POD for the tubular reactor problem (1), revealed by the numerical implementation and testing of the Galerkin POD ROM, is not a setback for the non-linear reduced order modeling approach discussed here, as our focus is on the interpolation method, not the specific basis used in the discretization.

3 An Entropy-Stable and Efficient Reduced Order Model (ROM) for the 3D Compressible Navier-Stokes Equations

Having formulated the "best" points interpolation of [15, 16] in the context of a non-linear Galerkin reduced order model, and demonstrated its application to a simple 1D non-linear convection-diffusion-reaction system (Section 2), let us now turn our attention to the equations of interest, namely the three-dimensional (3D) compressible Navier-Stokes equations. Following a discussion of the fluid variables, the governing equations and the boundary conditions (Section 3.1), we exhibit an entropy-stable inner product for the Galerkin projection step (Sections 3.2–3.4), and formulate the "best" points interpolation procedure outlined in Section 2.3 as it would be used to handle the non-linear terms present in these equations (Sections 3.5–3.6).

3.1 Notation and Governing Equations

In terms of the so-called conservation variables $\mathbf{U}$, the Navier-Stokes equations can be written as[5] (neglecting forces) [9]:

$$\mathbf{U}_{,t} + \mathbf{F}_{i,i} = \mathbf{F}^v_{i,i} + \mathbf{F}^h_{i,i} \tag{52}$$

where, in three-dimensions (3D):

$$\mathbf{U} \equiv \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{pmatrix} \equiv \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho e \end{pmatrix} \tag{53}$$

$$\mathbf{F}_i = u_i \mathbf{U} + p \begin{pmatrix} 0 \\ \delta_{1i} \\ \delta_{2i} \\ \delta_{3i} \\ u_i \end{pmatrix}, \quad \mathbf{F}^v_i = \begin{pmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ \tau_{3i} \\ \tau_{ij} u_j \end{pmatrix}, \quad \mathbf{F}^h_i = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -q_i \end{pmatrix} \tag{54}$$

for $i = 1, 2, 3$. $\mathbf{F}_i$ is known as the convective or Euler flux, $\mathbf{F}^v_i$ is the viscous flux, and $\mathbf{F}^h_i$ is the heat flux. The variables and parameters appearing in (53)–(54) are defined in Table 4. The specific heats are assumed to be positive constants. Moreover, we require that

$$\mu \geq 0, \quad \lambda + \frac{2}{3}\mu \geq 0, \quad \kappa \geq 0 \tag{55}$$

(52) is the conservative form of the 3D compressible Navier-Stokes equations. These equations can also be written in non-conservative form as

$$\mathbf{U}_{,t} + \mathbf{A}_i \mathbf{U}_{,i} = (\mathbf{K}_{ij} \mathbf{U}_{,j})_{,i} \tag{56}$$

[5] Note that we are employing the so-called Einstein notation, or implied summation on repeated indices, so that, e.g., $\mathbf{F}_{i,i} \equiv \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} + \frac{\partial \mathbf{F}_3}{\partial x_3}$.

Table 4: Fluid variables

Variable	Physical Meaning	Expression
ρ	fluid density	
u_i	fluid velocity in the i^{th} direction	
δ_{ij}	Kronecker delta	$\delta_{ij} = 1$ if $i = j$, $\delta_{ij} = 0$ otherwise
e	total energy density	$e = \imath + \frac{1}{2}u^2$
$\imath$	internal energy density	$\imath = c_v\theta$
θ	absolute temperature	
c_v	specific heat at constant volume	
c_p	specific heat at constant pressure	
γ	ratio of specific heats	$\gamma = c_p/c_v$
p	fluid pressure	$p = (\gamma-1)\rho\imath$
τ_{ij}	viscous stress	$\tau_{ij} = \lambda u_{k,k}\delta_{ij} + \mu(u_{i,j} + u_{j,i})$
λ, μ	viscosity coefficients	
q_i	heat flux	$q_i = -\kappa\theta_{,i}$
κ	conductivity	
η	thermodynamic entropy density per unit mass	
s	nondimensional entropy	$s \equiv \eta/c_v = \ln(p\rho^{-\gamma}) + \text{const}$ [Gibbs' equation]
t	time	
$\mathbf{x}$	position vector in Cartesian coordinates	$\mathbf{x}^T = (x_1, x_2, x_3)$

where $\mathbf{A}_i \equiv \mathbf{A}_i(\mathbf{U})$, $\mathbf{K}^v_{ij} \equiv \mathbf{K}^v_{ij}(\mathbf{U})$ and $\mathbf{K}^h_{ij} \equiv \mathbf{K}^h_{ij}(\mathbf{U})$ are defined by

$$\mathbf{F}_{i,i} = \mathbf{F}_{i,\mathbf{U}}\mathbf{U}_{,i} \equiv \mathbf{A}_i\mathbf{U}_{,i} \tag{57}$$

$$\mathbf{F}^v_i \equiv \mathbf{K}^v_{ij}\mathbf{U}_{,j} \tag{58}$$

$$\mathbf{F}^h_i \equiv \mathbf{K}^h_{ij}\mathbf{U}_{,j} \tag{59}$$

and

$$\mathbf{K}_{ij} \equiv \mathbf{K}^v_{ij} + \mathbf{K}^h_{ij} \tag{60}$$

Let us for now neglect the far-field boundary conditions, so that we consider only the solid wall boundary conditions, denoting the solid wall boundary of the domain Ω by $\partial\Omega_W \equiv \partial\Omega$. The relevant boundary conditions at the solid wall are:

$$\begin{array}{lll} \text{no slip BC:} & \mathbf{u} = \mathbf{0}, & \text{on } \partial\Omega_W \\ \text{adiabatic wall BC:} & \nabla\theta\cdot\mathbf{n} = 0, & \text{on } \partial\Omega_W \end{array} \tag{61}$$

This document (Section 3.4) also includes a discussion of the no-penetration boundary condition:

$$\text{no-penetration BC:} \quad \mathbf{u}\cdot\mathbf{n} = 0, \quad \text{on } \partial\Omega_W \tag{62}$$

which it may be desirable to implement, for instance, if the basis functions employed do not satisfy the no-slip condition at the wall.

3.2 Clausius-Duhem Inequality, Entropy Variables and Symmetrization of the Navier-Stokes Equations

In designing a Galerkin Reduced Order Model (ROM) for the compressible Navier Stokes equations (52), we are interested in defining an inner product in which the Galerkin projection will be stable. As discussed in [5, 9], stability can be ensured by the energy method.

3.2.1 Clausius-Duhem Inequality

For the full (non-linear) Euler or Navier-Stokes equations, the energy method is closely tied to the second law of thermodynamics, or the Clausius-Duhem inequality, namely

$$\frac{d}{dt}\int_{\Omega}\rho\eta\, d\Omega \geq -\int_{\partial\Omega_W}\frac{q_i n_i}{\theta}dS \tag{63}$$

where η is the thermodynamic entropy density per unit mass (Table 4). (63) essentially states that the entropy of the system is non-decreasing. For (52), energy estimates, or the satisfaction of the entropy inequality (63), imply that the semi-discrete solutions possess stability properties akin to those of the exact solutions of the governing equations [5, 9]. We will call solutions that satisfy (63) "entropy-stable". Our aim here is to develop a transformation (symmetrization) and define an inner product such that the Clausius-Duhem inequality (63) is necessarily satisfied for the Galerkin ROM we will build for the compressible Navier-Stokes equations (52) with boundary conditions (61).

3.2.2 Entropy Variables

To develop a Clausius-Duhem inequality-preserving Galerkin projection of the equations (56), let us introduce a change of variables $\mathbf{U} \rightarrow \mathbf{V}$:

$$\mathbf{U} = \mathbf{U}(\mathbf{V}) \tag{64}$$

We will refer to $\mathbf{V}$ as the "entropy variables". In terms of the entropy variables $\mathbf{V}$, the equations of interest (56) are:

$$\mathbf{A}_0\mathbf{V}_{,t} + \tilde{\mathbf{A}}_i\mathbf{V}_{,i} - (\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j})_{,i} = \mathbf{0} \tag{65}$$

where[6]

$$\mathbf{A}_0 \equiv \mathbf{U}_{,\mathbf{V}} \tag{66}$$

$$\tilde{\mathbf{A}}_i \equiv \mathbf{A}_i\mathbf{A}_0 \tag{67}$$

$$\tilde{\mathbf{K}}_{ij} \equiv \mathbf{K}_{ij}\mathbf{A}_0 \tag{68}$$

It is well-known that the matrices $\mathbf{A}_i$ in (56) are non-symmetric. However, it is also well-known that all linear combinations of the $\mathbf{A}_i$ possess real eigenvalues and a complete set of eigenvectors, meaning $\mathbf{U}_{,t} + \mathbf{A}_i\mathbf{U}_{,i} = \mathbf{0}$ constitutes a hyperbolic system of conservation laws. We seek a change of variables (64) such that:

1. The matrices $\mathbf{A}_0$ and $\tilde{\mathbf{A}}_i$ are symmetric, and
2. The matrix

$$\tilde{\mathbf{K}} \equiv \begin{pmatrix} \tilde{\mathbf{K}}_{11} & \tilde{\mathbf{K}}_{12} & \tilde{\mathbf{K}}_{13} \\ \tilde{\mathbf{K}}_{21} & \tilde{\mathbf{K}}_{22} & \tilde{\mathbf{K}}_{23} \\ \tilde{\mathbf{K}}_{31} & \tilde{\mathbf{K}}_{32} & \tilde{\mathbf{K}}_{33} \end{pmatrix} \tag{69}$$

   is symmetric positive semi-definite.

If the transformation (64) is defined such that these properties hold, the resulting system in the entropy variables will be a symmetric hyperbolic system.

3.2.3 Generalized Entropy Functions and Entropy Fluxes

Following the symmetrization approaches of [5, 9], we will define the change of variables (64) with the help of so-called generalized entropy functions. A generalized entropy function $H \equiv H(\mathbf{U})$ is by definition a function that satisfies the following two conditions [9]:

1. H is convex[7].

[6] The reader is referred to Section 5.1 of the Appendix for explicit expressions of the symmetrized matrices (66)–(68).

[7] The convexity of H is equivalent to the positive-definiteness of $\mathbf{A}_0$, since $\mathbf{A}_0^{-1} = \mathbf{V}_{,\mathbf{U}} = H_{,\mathbf{UU}}$.

2. There exist scalar-valued function $\sigma_i \equiv \sigma_i(\mathbf{U})$, $i = 1,2,3$, referred to as entropy fluxes, such that

$$H_{,\mathbf{U}}\mathbf{A}_i = \sigma_{i,\mathbf{U}} \tag{70}$$

The following theorems, quoted from [6], delineate the relationship between symmetric hyperbolic systems and generalized entropy functions, and will be employed in our symmetrization of the equations (57):

Theorem 3.2.1 (Mock). *A hyperbolic system of conservation laws possessing a generalized entropy function becomes symmetric under the change of variables*

$$\mathbf{V}^T = H_{,\mathbf{U}} \tag{71}$$

Theorem 3.2.2 (Godunov). *If a hyperbolic system can be symmetrized by introducing a change of variables, then a generalized entropy function and corresponding entropy fluxes exist for this system.*

3.2.4 Entropy Flux for the Compressible Navier-Stokes Equations (52)

It is shown in [6, 9] that for the compressible Navier-Stokes equations (56), appropriate choices for the entropy flux and entropy function are

$$\sigma_i = Hu_i, \qquad H = -\rho g(s) \equiv -\rho s \tag{72}$$

respectively. Here s is the non-dimensional entropy, $s = \eta/c_v$ (Table 4), which satisfies the well-known Gibbs equation $s = \ln(p\rho^{-\gamma}) + \text{const.}$

With the choice of affine entropy flux (72), the transformation $\mathbf{U} \to \mathbf{V}$ (71) is given by

$$\mathbf{V} = \frac{1}{\rho_1}\begin{pmatrix} -U_5 + \rho_1(\gamma + 1 - s) \\ U_2 \\ U_3 \\ U_4 \\ -U_1 \end{pmatrix} \tag{73}$$

where

$$s = \ln\left[\frac{(\gamma-1)\rho_1}{U_1^\gamma}\right] \tag{74}$$

$$\rho_1 = U_5 - \frac{1}{2U_1}(U_2^2 + U_3^2 + U_4^2) \tag{75}$$

The inverse mapping $\mathbf{V} \to \mathbf{U}$ is given by

$$\mathbf{U} = \rho_1\begin{pmatrix} -V_5 \\ V_2 \\ V_3 \\ V_4 \\ 1 - \frac{1}{2V_5}(V_2^2 + V_3^2 + V_4^2) \end{pmatrix} \tag{76}$$

where

$$\rho_1 = \left[\frac{\gamma-1}{(-V_5)^\gamma}\right]^{1/(\gamma-1)} \exp\left(\frac{-s}{\gamma-1}\right) \tag{77}$$

$$s = \gamma - V_1 + \frac{1}{2V_5}(V_2^2 + V_3^2 + V_4^2) \tag{78}$$

3.2.5 Homogeneous vs. Inhomogeneous Entropy Fluxes

We conclude the discussion by calling attention to the fact that the affine entropy flux (72) is *not* homogeneous. To be called a homogeneous flux function [5], H must be selected such that:

$$\mathbf{U}_{,\mathbf{V}}\mathbf{V} = \beta\mathbf{U} \tag{79}$$

$$\tilde{\mathbf{F}}_{i,\mathbf{V}}\mathbf{V} = \beta\tilde{\mathbf{F}}_i \tag{80}$$

for some $\beta \in \mathbb{R}$, where $\tilde{\mathbf{F}}_i \equiv \mathbf{F}_i(\mathbf{V}(\mathbf{U}))$ (the Euler fluxes in the transformed entropy variables). As shown in [6], the *viscous* terms in the Navier-Stokes equations will be symmetric and positive definite after symmetrization for any member $H = -\rho g(s)$ of Harten's generalized entropy functions, in particular the following family of exponential *homogeneous* flux functions:

$$h(s) = Ke^{\kappa s} = K(p\rho^{-\gamma}), \qquad K, \kappa \neq 0 \tag{81}$$

However, as proven in Section 5.5 of the Appendix, if the heat flux term $\mathbf{F}^h_{i,i}$ is present in the equations (52), the only way for the augmented heat flux matrix (69) to remain positive semi-definite is if H is affine in s, i.e., if H has the form (72). It is for this reason that we have selected the *inhomogeneous* entropy flux function (72) for the compressible Navier-Stokes equations (52) instead of the *homogeneous* flux function (81). The latter could be used for the Euler equations or the Navier-Stokes equations with $\mathbf{F}^h_{i,i}$. In our case, since $\mathbf{F}^h_{i,i} \neq \mathbf{0}$, we select (72) to obtain the entropy-stability result in Theorem 3.3.1.

3.3 Entropy Stable Galerkin Projection of the Symmetrized Compressible Navier-Stokes Equations (65) with Boundary Conditions (61)

Let us now examine the stability of the Galerkin projection in the $L_2(\Omega)$ inner product of the symmetrized compressible Navier-Stokes equations (65) with boundary conditions (61). We will say that the Galerkin projection is "entropy-stable" if it satisfies the Clausius-Duhem entropy inequality (63), or the second law of thermodynamics. Per the discussion in [5, 9], we aim to show that the change of variables (73) is such that when the transformed equations (65) are projected onto a POD mode, the Clausius-Duhem inequality is respected *ab initio* for all numerical solutions.

Assume the entropy variables have been expanded in a vector basis $\{\boldsymbol{\phi}_i\}_{i=1}^M \in \mathbb{R}^5$:

$$\mathbf{V}(\mathbf{x},t) \approx \mathbf{V}_M(\mathbf{x},t) = \sum_{m=1}^{M} a_m(t)\boldsymbol{\phi}_m(\mathbf{x}) \tag{82}$$

where the $a_m(t)$ are the modal amplitudes (or ROM coefficients) to be solved for. Assume the basis is orthonormal in the $L_2(\Omega)$ inner product, so that $(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \delta_{ij}$ for all $i, j = 1, \ldots, M$.

Theorem 3.3.1. *Consider the symmetrized compressible 3D Navier-Stokes equations* (65) *in an open bounded domain* $\Omega \subset \mathbb{R}^3$*, with the no-slip and adiabatic wall boundary condition* (61) *on the boundary* $\partial\Omega_W$*. Define the transformation* $\mathbf{U} \to \mathbf{V}$ *given by the entropy flux* (72)*, so that the relationship between* $\mathbf{U}$ *and the entropy variables* $\mathbf{V}$ *is* (73)*. Then the Galerkin projection onto a POD mode* $\boldsymbol{\phi}_j$ *of* (65) *with boundary conditions* (61) *in the* $L^2(\Omega)$ *inner product is "entropy stable" (i.e.., satisfies the entropy estimate* (63)*) if the POD modes* $\boldsymbol{\phi}_j$ *satisfy the no-slip condition on* $\partial\Omega_W$*, i.e., if*

$$\phi_j^2 = \phi_j^3 = \phi_j^4 = 0 \tag{83}$$

for $j = 1, \ldots, M$ *where* ϕ_j^i *denotes the* i^{th} *component of* $\boldsymbol{\phi}_j$ *for* $i = 1, \ldots, 5$*.*

Proof. Let us work out the projection of each of the terms at (65), one at a time. Premultiplying (65) by $\mathbf{V}^T$ and integrating over Ω, we have:

$$\begin{aligned} \int_\Omega \mathbf{V}^T\mathbf{A}_0\mathbf{V}_{,t}d\Omega &= \int_\Omega H_{,\mathbf{U}}\mathbf{U}_{,\mathbf{V}}\mathbf{V}_{,t}d\Omega \\ &= \int_\Omega H_{,\mathbf{U}}\mathbf{U}_{,t}d\Omega \\ &= \int_\Omega H_{,t}d\Omega \end{aligned} \tag{84}$$

Note that

$$\mathbf{V}^T\tilde{\mathbf{A}}_i = (H_{,\mathbf{U}}\mathbf{A}_i)\mathbf{A}_0 = \sigma_{i,\mathbf{U}}\mathbf{U}_{,\mathbf{V}} = \sigma_{i,\mathbf{V}} \tag{85}$$

Now, for the convection term:

$$\int_\Omega \mathbf{V}^T\tilde{\mathbf{A}}_i\frac{\partial\mathbf{V}}{\partial x_i}d\Omega = \int_\Omega \underbrace{\sigma_{i,\mathbf{V}}\frac{\partial\mathbf{V}}{\partial x_i}}_{\sigma_{i,i}}d\Omega = \int_\Omega (Hu_i)_{,i}d\Omega \tag{86}$$

Moving on to the diffusion term:

$$\begin{aligned}
\textstyle\int_\Omega \mathbf{V}^T(\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j})_{,i}d\Omega &= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_\Omega(\mathbf{V}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j})_{,i}d\Omega \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}dS \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T\mathbf{K}_{ij}\mathbf{n}_i\mathbf{A}_0\mathbf{V}_{,j}dS \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T(\mathbf{K}_{ij}^v+\mathbf{K}_{ij}^h)\mathbf{n}_i\mathbf{U}_{,\mathbf{V}}\mathbf{V}_{,j}dS \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T(\mathbf{K}_{ij}^v+\mathbf{K}_{ij}^h)\mathbf{n}_i\mathbf{U}_{,\mathbf{V}}dS \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T(\mathbf{F}_i^v+\mathbf{F}_i^h)\mathbf{n}_i dS \\
&= -\textstyle\int_\Omega \mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \int_{\partial\Omega_W}\mathbf{V}^T\mathbf{F}_i^v n_i dS + \frac{1}{c_v}\int_{\partial\Omega_W}\frac{\mathbf{q}_i\mathbf{n}_i}{\theta}dS
\end{aligned} \tag{87}$$

The integrand in the first boundary integral in (87) becomes, after the application of the no-slip condition (see (172))

$$[\mathbf{F}_i^v n_i]^{ns} = \frac{\mu}{V_5^2}\begin{pmatrix} 0 \\ (-V_5V_{i+1,1}+V_{i+1}V_{5,1}-V_5V_{2,i}+V_2V_{5,i})n_i \\ (-V_5V_{i+1,2}+V_{i+1}V_{5,2}-V_5V_{3,i}+V_3V_{5,i})n_i \\ (-V_5V_{i+1,3}+V_{i+1}V_{5,3}-V_5V_{4,i}+V_4V_{5,i})n_i \\ 0 \end{pmatrix} + \lambda\left[\frac{-V_5V_{i+1,i}+V_{i+1}V_{5,i}}{V_5^2}\right]\begin{pmatrix} 0 \\ n_1 \\ n_2 \\ n_3 \\ 0 \end{pmatrix} \tag{88}$$

Let $\boldsymbol{\phi}_j \in \mathbb{R}^5$ be a POD mode for the primal unknown field in the entropy variables, $\mathbf{V}$, and assume that $\boldsymbol{\phi}_j$ satisfies the no-slip condition (e.g., assume $\phi_j^2, \phi_j^3, \phi_j^4$ on $\partial\Omega_W$ is zero-ed out *a posteriori* in the implementation to ensure that it satisfies no-slip). Then, it follows from (88) that $[\boldsymbol{\phi}_j^T\mathbf{F}_i^v n_i]^{ns} = 0$ necessarily for all j, meaning $[\mathbf{V}^T\mathbf{F}_i^v n_i]^{ns} = 0$.

Putting (84), (86) and (87) together, we obtain:

$$\begin{aligned}
\tfrac{1}{c_v}\textstyle\int_\Omega(\rho\eta)_{,t}d\Omega &= \textstyle\int_\Omega c_v\mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \frac{1}{c_v}\int_\Omega\left[-(Hu_i)_{,i} - \left(\tfrac{q_i}{\theta}\right)_{,i}\right]d\Omega \\
&= \textstyle\int_\Omega c_v\mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega + \frac{1}{c_v}\int_{\partial\Omega_W}\left[-H \quad \underbrace{u_i n_i}_{=0\text{ (by no-slip BC)}} \quad -\left(\tfrac{q_i}{\theta}\right)n_i\right]dS \\
&= \textstyle\int_\Omega c_v\mathbf{V}_{,i}^T\tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}d\Omega - \frac{1}{c_v}\int_{\partial\Omega_W}\underbrace{\left(\frac{q_i n_i}{\theta}\right)}_{=0\text{ (by adiabatic wall BC)}}dS \\
&\geq 0
\end{aligned} \tag{89}$$

or

$$\frac{d}{dt}\int_\Omega \rho\eta d\Omega \geq 0 \tag{90}$$

which implies non-decreasing entropy (63), and therefore entropy-stability of the Galerkin projection. □

3.4 Weak Formulation and Implementation of Boundary Conditions (61) and (62)

Let us now formulate a weak implementation of the boundary conditions (61), using the viscous fluxes to implement the no-slip condition. We also formulate the implementation of the no-penetration boundary condition (62) using the

convection term, which may be required for the numerics if the basis functions $\boldsymbol{\phi}_m$ do *not* satisfy the no-slip condition on $\partial\Omega_W$. Projecting (56) onto the mode $\boldsymbol{\phi}_m$, gives:

$$\begin{aligned}\int_\Omega \boldsymbol{\phi}_m^T \mathbf{A}_0 \mathbf{V}_{,t} d\Omega &= -\int_\Omega \boldsymbol{\phi}_m^T \tilde{\mathbf{A}}_i \mathbf{V}_{,i} d\Omega + \int_\Omega \boldsymbol{\phi}_m^T ([\tilde{\mathbf{K}}_{ij}^v + \tilde{\mathbf{K}}_{ij}^h]\mathbf{V}_{,j})_{,i} d\Omega \\ &= \int_\Omega (\boldsymbol{\phi}_m^T \tilde{\mathbf{A}}_i)_{,i} \mathbf{V} d\Omega - \int_\Omega \boldsymbol{\phi}_{m,i}^T \tilde{\mathbf{K}}_{ij} \mathbf{V}_{,j} d\Omega \\ &\underbrace{-\int_{\partial\Omega_W} \boldsymbol{\phi}_m^T [\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np} dS}_{=I_m^{np}} + \underbrace{\int_{\partial\Omega_W} \boldsymbol{\phi}_m^T [\tilde{\mathbf{K}}_{ij}^v n_i \mathbf{V}_{,j}]^{ns} dS}_{=I_m^{ns}} + \underbrace{\int_{\partial\Omega_W} \boldsymbol{\phi}_m^T [\tilde{\mathbf{K}}_{ij}^h n_i \mathbf{V}_{,j}]^{ad} dS}_{=I_m^{ad}}\end{aligned} \tag{91}$$

From Sections 5.2–5.4 of the Appendix, we have that:

$$[\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np} = \begin{pmatrix} 0 \\ -\rho \imath s n_1 \\ -\rho \imath s n_2 \\ -\rho \imath s n_3 \\ 0 \end{pmatrix} \tag{92}$$

$$[\tilde{\mathbf{K}}_{ij}^v n_i \mathbf{V}_{,j}]^{ns} = \frac{\mu}{V_5^2} \begin{pmatrix} 0 \\ (-V_5 V_{i+1,1} + V_{i+1} V_{5,1} - V_5 V_{2,i} + V_2 V_{5,i}) n_i \\ (-V_5 V_{i+1,2} + V_{i+1} V_{5,2} - V_5 V_{3,i} + V_3 V_{5,i}) n_i \\ (-V_5 V_{i+1,3} + V_{i+1} V_{5,3} - V_5 V_{4,i} + V_4 V_{5,i}) n_i \\ 0 \end{pmatrix} + \lambda \left[\frac{-V_5 V_{i+1,i} + V_{i+1} V_{5,i}}{V_5^2} \right] \begin{pmatrix} 0 \\ n_1 \\ n_2 \\ n_3 \\ 0 \end{pmatrix} \tag{93}$$

and

$$[\tilde{\mathbf{K}}_{ij}^h n_i \mathbf{V}_{,j}]^{ad} = \mathbf{0} \tag{94}$$

Denoting

$$[\phi_m]_n \equiv \phi_m^2 n_1 + \phi_m^3 n_2 + \phi_m^4 n_3 \tag{95}$$

we obtain the following expressions for the boundary integrals in (91) (Table 5).

Table 5: Boundary integrals arising from the weak implementation of the BCs (61)

Boundary Integral	Expression
I_m^{np}	$\int_{\partial\Omega_W} \left[\frac{\gamma-1}{(-V_5)^2}\right]^{1/(\gamma-1)} \exp\left(\frac{-\gamma+V_1-\frac{1}{2V_5}(V_2^2+V_3^2+V_4^2)}{\gamma-1}\right)[\gamma - V_1 + \frac{1}{2V_5}(V_2^2+V_3^2+V_4^2)][\phi_m]_n dS$
I_m^{ns}	$\int_{\partial\Omega_W} \left[-\frac{\mu}{V_5^2}(V_5 V_{i+1,j} - V_{i+1} V_{5,j} + V_5 V_{j+1,i} - V_{j+1} V_{5,i}) n_i \phi^{m+1} - \lambda \left(\frac{V_5 V_{i+1,i} - V_{i+1} V_{5,i}}{V_5^2}\right)[\phi_m]_n\right] dS$
I_m^{ad}	0

Note that if the POD modes $\boldsymbol{\phi}_m$ satisfy the no-slip condition on $\partial\Omega_W$, i.e., $\phi_m^2 = \phi_m^3 = \phi_m^4 = 0$, then the integrals I_m^{np} and I_m^{ns} in Table 5 are identically 0: $I_m^{np} = I_m^{ns} \equiv 0$.

The non-linearity in the full Navier-Stokes equations (52) is in the advection term, or Euler fluxes $\mathbf{A}_i$. Note, however, that the diffusive terms in the entropy variable analog of (52), namely (65), are also non-linear, due to the fact that the symmetrizing matrix (Jacobian) $\mathbf{A}_0 \equiv \mathbf{U}_{,\mathbf{V}}$ is a function of $\mathbf{V}$. Hence, *all* the symmetrized matrices, namely $\tilde{\mathbf{A}}_i$ and $\tilde{\mathbf{K}}_{ij}$ will be non-linear in $\mathbf{V}$; in the ROM with boundary conditions, the boundary integrals (Table 5) will contain non-linearities as well if the basis functions do not satisfy no-slip. Moreover, since $\mathbf{A}_0 \equiv \mathbf{A}_0(\mathbf{V})$, while one has that $(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) = \delta_{ij}$ for any two basis functions $\boldsymbol{\phi}_j, \boldsymbol{\phi}_i$,

$$(\boldsymbol{\phi}_i, \mathbf{A}_0 \boldsymbol{\phi}_j) \neq \delta_{ij} \tag{96}$$

A consequence of (96) is a mass matrix will appear in the semi-discrete ROM to be advanced forward in time (see e.g., (130)).

Introducing the shorthand, for $\mathbf{V}_1, \mathbf{V}_2 \in \mathbb{R}^5$:

$$(\mathbf{V}_1, \mathbf{V}_2) \equiv \int_\Omega \mathbf{V}_1^T \mathbf{V}_2 d\Omega, \qquad \langle \mathbf{V}_1, \mathbf{V}_2 \rangle_{\partial\Omega_W} \equiv \int_{\partial\Omega_W} \mathbf{V}_1^T \mathbf{V}_2 dS \tag{97}$$

the governing equations (65) projected onto a POD mode $\boldsymbol{\phi}_m$ are

$$(\boldsymbol{\phi}_m, \mathbf{A}_0\mathbf{V}_{,t}) - \left((\boldsymbol{\phi}_m\tilde{\mathbf{A}}_i)_{,i}, \mathbf{V}\right) + (\boldsymbol{\phi}_{m,i}, \tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}) \underbrace{-\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np}\rangle_{\partial\Omega_W}}_{=I_m^{np}} + \underbrace{\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{K}}^v_{ij} n_i \mathbf{V}_{,j}]^{ns}\rangle_{\partial\Omega_W}}_{=I_m^{ns}} + \underbrace{\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{K}}^h_{ij} n_i \mathbf{V}_{,j}]^{ad}\rangle_{\partial\Omega_W}}_{=I_m^{ad}=0} = 0 \tag{98}$$

or (setting $I_{ad} = 0$; see Table 5)

$$(\boldsymbol{\phi}_m, \mathbf{A}_0\mathbf{V}_{,t}) - \left(\boldsymbol{\phi}_{m,i}, \tilde{\mathbf{A}}_i\mathbf{V}\right) - \left(\boldsymbol{\phi}_m, \tilde{\mathbf{A}}_{i,i}\mathbf{V}\right) + (\boldsymbol{\phi}_{m,i}, \tilde{\mathbf{K}}_{ij}\mathbf{V}_{,j}) \underbrace{-\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np}\rangle_{\partial\Omega_W}}_{=I_m^{np}} + \underbrace{\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{K}}^v_{ij} n_i \mathbf{V}_{,j}]^{ns}\rangle_{\partial\Omega_W}}_{=I_m^{ns}} = 0 \tag{99}$$

Substituting the modal expansion (82) into (99), one obtains

$$\begin{aligned}\textstyle\sum_{n=1}^M (\boldsymbol{\phi}_m, [\mathbf{A}_0]_M\boldsymbol{\phi}_n)\,\dot{a}_n \quad &= \left(\boldsymbol{\phi}_{m,i}, [\tilde{\mathbf{A}}_i]_M\mathbf{V}_M\right) + \left(\boldsymbol{\phi}_m, [\tilde{\mathbf{A}}_{i,i}]_N\mathbf{V}_M\right) - (\boldsymbol{\phi}_{m,i}, [\tilde{\mathbf{K}}_{ij}]_M\mathbf{V}_{M,j}) + \langle\boldsymbol{\phi}_m, [\tilde{\mathbf{A}}_i n_i\mathbf{V}]^{np}_M\rangle_{\partial\Omega_W} \\ &\quad -\langle\boldsymbol{\phi}_m, [\tilde{\mathbf{K}}^v_{ij} n_i \mathbf{V}_{,j}]^{ns}_M\rangle_{\partial\Omega_W}\end{aligned} \tag{100}$$

where $[\mathbf{A}_0]_M \equiv \mathbf{A}_0(\mathbf{V}_M) = \mathbf{A}_0\left(\sum_{n=1}^M a_n(t)\boldsymbol{\phi}_n\right)$ and similarly for the other matrices with "M" subscripts in (100).

All the terms in the projected equations (99) contain non-linearities[8], including the term on the left-hand side. We will denote the non-linear terms as follows:

$$[\mathbf{f}_0(\mathbf{V}_M)]_n = [\mathbf{A}_0]_M\boldsymbol{\phi}_n, \qquad n = 1, ..., M \tag{101}$$

$$\mathbf{f}_i(\mathbf{V}_M) \equiv [\tilde{\mathbf{A}}_i]_M\mathbf{V}_M, \qquad i = 1,2,3 \tag{102}$$

$$\mathbf{f}_4(\mathbf{V}_M) \equiv [\tilde{\mathbf{A}}_{i,i}]_M\mathbf{V}_M \tag{103}$$

$$\mathbf{f}_i(\mathbf{V}_M) \equiv [\tilde{\mathbf{K}}_{ij}]_M\mathbf{V}_{M,j}, \qquad i = 5,6,7 \tag{104}$$

$$\mathbf{f}_8(\mathbf{V}_M) \equiv [\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np}_M \tag{105}$$

$$\mathbf{f}_9(\mathbf{V}_M) \equiv [\tilde{\mathbf{K}}^v_{ij} n_i \mathbf{V}_{,j}]^{ns}_M \tag{106}$$

Then (100) takes the form (for $i = 1,2,3$)

$$\begin{aligned}\textstyle\sum_{n=1}^M (\boldsymbol{\phi}_m, [\mathbf{f}_0(\mathbf{V}_M)]_n)\,\dot{a}_n \quad &= \left(\boldsymbol{\phi}_{m,i}, \mathbf{f}_i(\mathbf{V}_M)\right) + (\boldsymbol{\phi}_m, \mathbf{f}_4(\mathbf{V}_M)) - (\boldsymbol{\phi}_{m,i}, \mathbf{f}_{i+4}(\mathbf{V}_M)) + \langle\boldsymbol{\phi}_m, \mathbf{f}_8(\mathbf{V}_M)\rangle_{\partial\Omega_W} \\ &\quad -\langle\boldsymbol{\phi}_m, \mathbf{f}_9(\mathbf{V}_M)\rangle_{\partial\Omega_W}\end{aligned} \tag{107}$$

for $t \in (0,T]$ subject to the initial condition $\mathbf{V}(0,\mathbf{x}) = \mathbf{V}_0(\mathbf{x})$. Once discretized in time, (107) will yield a non-linear discrete system of equations that can be advanced in time using an explicit time integration scheme, or by combining an implicit scheme with Newton's method at each time step. Note that, unlike in the case of a ROM for linear equations, the left-hand side of (107) will contain a mass matrix that will need to be inverted during the time-integration of the ROM.

3.5 "Best Points" Interpolation for Non-Linear Projected Terms

As in the 1D tubular reactor problem of Section 2.3, applying the standard Galerkin reduced-order model to (107) is inefficient due to the presence of the non-linear terms. To recover efficiency, let us develop the coefficient function approximation (Section 2.3) to the non-linear terms in this expression.

As outlined in Algorithm 1, one begins by computing snapshots K for the primal unknown field $\mathbf{V}$, at K different times t_k:

$$\mathscr{S}^{\mathbf{V}} \equiv \{\boldsymbol{\xi}^{\mathbf{V}}_k(\mathbf{x}) = \mathbf{V}^k_h(\mathbf{x}) : 1 \le k \le K\} \tag{108}$$

[8] As discussed in Section 3.4, if the POD modes $\boldsymbol{\phi}_m$ satisfy the no-slip condition at $\partial\Omega_W$ the boundary integrals I_m^{ns} and I_m^{np} vanish.

Given this set of snapshots of the flow field, one then computes snapshots for each of the non-linear functions in (101) – (106):

$$\mathscr{S}^{[\mathbf{f}_0]_n} \equiv \{\boldsymbol{\xi}_k^{[\mathbf{f}_0]_n}(\mathbf{x}) = [\mathbf{f}_0(\mathbf{V}_h^k(\mathbf{x}))]_n : 1 \leq k \leq K\}, \qquad n = 1,...,M \tag{109}$$

$$\mathscr{S}^{\mathbf{f}_j} \equiv \{\boldsymbol{\xi}_k^{\mathbf{f}_j}(\mathbf{x}) = \mathbf{f}_j(\mathbf{V}_h^k(\mathbf{x})) : 1 \leq k \leq K\}, \qquad j = 1,...,9 \tag{110}$$

From these snapshots, one solves for the "best" interpolation points for each of the non-linear functions (101)–(106), denoted here by:

$$\{\mathbf{z}_m^{[\mathbf{f}_0]_n}\}_{m=1}^M : \quad \text{"best" (or any) interpolation points for } [\mathbf{f}_0]_n, n = 1,...,M \tag{111}$$

$$\{\mathbf{z}_m^{\mathbf{f}_j}\}_{m=1}^M : \quad \text{"best" (or any) interpolation points for } \mathbf{f}_j, j = 1,...,9 \tag{112}$$

following the approach outlined above in Section 2.3 and in the journal article [16].

We note that the main difference between the non-linear functions that appear in the projected Navier-Stokes equations (107) and the projected equations for the tubular reactor (11) is that the non-linear functions in the former are vector-valued. However, in practice, this poses no difficulty for the solution procedure of Section 2.3, as this exact procedure can be applied to each *component* of each of the non-linear vector-valued function in (101)–(106). For concreteness, let $\mathbf{f}_j \in \mathbb{R}^5$ be any of the vector-valued functions in (101)–(106), and let f_j^i denote the i^{th} component of $\mathbf{f}_j$ for $j = 0,1...,9$, $i = 1,...,5$. Then, each of the components of each of the functions $\mathbf{f}_j$ can be expanded in an orthonormal (scalar) basis as, denoted here by $\{\phi_m^{f_j^i}\}_{m=1}^M$. Now, we can define the best approximations of the elements in the snapshot set as:

$$[f_j^i]_M^*(\mathbf{V}_h^k) = \sum_{m=1}^M \alpha_m^{f_j^i} \phi_m^{f_j^i}(\mathbf{x}), \qquad 1 \leq k \leq K \tag{113}$$

where

$$\alpha_m^{f_j^i} = (\phi_m, f_j^i(\mathbf{V}_h^k(\cdot))), \qquad m = 1,...,M, 1 \leq k \leq K \tag{114}$$

for $i = 1,...,5$, $j = 0,1,...,9$. Now, the interpolation points for each component of each nonlinear function $\{\mathbf{z}_m^{f_j^i}\}_{m=1}^M \in \Omega \subset \mathbb{R}^3$ are defined as the solution to the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{z}_1,...,\mathbf{z}_M \in \Omega} \left|\left| [f_j^i]_M^*(\cdot) - \textstyle\sum_{m=1}^M \beta_m^{f_j^i}(\mathbf{z}_1,...,\mathbf{z}_M)\phi_m^{f_j^i} \right|\right|^2 \\ &\textstyle\sum_{n=1}^M \phi_n^{f_j^i}(\mathbf{z}_m)\beta_n^{f_j^i}(\mathbf{z}_1,...,\mathbf{z}_M) = f_j^i(\mathbf{z}_m), \qquad 1 \leq m \leq M \end{aligned} \tag{115}$$

Substituting (113) into (115) and invoking the orthonormality of the $\{\phi_m^{f_j^i}\}_{m=1}^M$, we obtain:

$$\begin{aligned} &\min_{\mathbf{z}_1,...,\mathbf{z}_M \in \Omega} \textstyle\sum_{m=1}^M (\alpha_m^{f_j^i} - \beta_m^{f_j^i}(\mathbf{z}_1,...,\mathbf{z}_M))^2 \\ &\textstyle\sum_{n=1}^M \phi_n^{f_j^i}(\mathbf{z}_m)\beta_n^{f_j^i}(\mathbf{z}_1,...,\mathbf{z}_M) = f_j^i(\mathbf{z}_m), \qquad 1 \leq m \leq M \end{aligned} \tag{116}$$

i.e., the set of points $\{\mathbf{z}_m^{f_j^i}\}_{m=1}^M$ is determined to minimize the average error between the interpolants $[f_j^i]_M(\cdot)$ and the best approximations $[f_j^i]_M^*(\cdot)$. Comparing the optimization problems (116) and (25), one can see that these are identical, with the general function f in (25) replaced by f_j^i, the i^{th} component of $\mathbf{f}_j$, one of the non-linear functions in (101)–(106), and so we refer the reader to Section 2.3 for details of the solution procedure for the "best" (or, hierarchical, if desired) interpolation points.

Given a set of interpolation points $\mathbf{z}_m^{f_j^i}$ for f_j^i, one can define the cardinal function $\{\psi_m^{f_j^i}\}$ for f_j^i by

$$\boldsymbol{\phi}_M^{f_j^i}(\mathbf{x}) = \mathbf{A}^{f_j^i} \boldsymbol{\psi}_M^{f_j^i}(\mathbf{x}) \tag{117}$$

where $\boldsymbol{\phi}_M^{f_j^i}(\mathbf{x}) = (\phi_1^{f_j^i}(\mathbf{x}), ..., \phi_M^{f_j^i}(\mathbf{x}))^T$ and $\boldsymbol{\psi}_M^{f_j^i}(\mathbf{x}) = (\psi_1^{f_j^i}(\mathbf{x}), ..., \psi_M^{f_j^i}(\mathbf{x}))^T$, and $A_{mn}^{f_j^i} = \phi_n^{f_j^i}(\mathbf{z}_m^{f_j^i})$. As before, the (scalar) cardinal functions $\psi_m^{f_j^i}$ satisfy $\psi_m^{f_j^i}(\mathbf{z}_n^{f_j^i}) = \delta_{mn}$. Given the interpolation points $\{\mathbf{z}_m^{f_j^i}\}_{m=1}^M$ and cardinal functions $\{\psi_m^{f_j^i}\}$ (117), one can approximate the i^{th} component ($i = 1, ..., 5$) of $\mathbf{f}_j$ (101) – (106) by

$$[f_0^i(\mathbf{V})]_n \approx [[f_0^i]_M(\mathbf{V})]_n = \sum_{m=1}^{M} [f_0^i]_n(\mathbf{V}(\mathbf{z}_m^{[f_0^i]_n}))\psi_m^{[f_0^i]_n} \in \mathbb{R}, \qquad n = 1, ..., M \tag{118}$$

$$f_j^i(\mathbf{V}) \approx [f_j^i]_M(\mathbf{V}) = \sum_{m=1}^{M} f_j^i(\mathbf{V}(\mathbf{z}_m^{f_j^i}))\psi_m^{f_j^i} \in \mathbb{R}, \qquad i = 1, .., 5, j = 1, ..., 9 \tag{119}$$

3.6 Reduced Order Approximation to the Symmetrized Compressible Navier-Stokes Equations with Interpolation

Our reduced-order approximation, (107) but now with interpolation, is obtained from (118) and (119) and takes the following form

$$\sum_{n=1}^{M} (\boldsymbol{\phi}_m, [\mathbf{f}_0(\mathbf{a}_M)]_n)\dot{a}_n = \left(\boldsymbol{\phi}_{m,i}, \mathbf{f}_i(\mathbf{a}_M)\right) + (\boldsymbol{\phi}_m, \mathbf{f}_4(\mathbf{a}_M)) - (\boldsymbol{\phi}_{m,i}, \mathbf{f}_{i+4}(\mathbf{a}_M)) - \langle \boldsymbol{\phi}_m, \mathbf{f}_8(\mathbf{a}_M)\rangle_{\partial\Omega_W} - \langle \boldsymbol{\phi}_m, \mathbf{f}_9(\mathbf{a}_M)\rangle_{\partial\Omega_W} \tag{120}$$

for $m = 1, ..., M$, where $\mathbf{a}_M^T \equiv (a_1, ..., a_M) \in \mathbb{R}^M$ and

$$[f_0^i]_n \approx [f_0^i(\mathbf{a}_M)]_n = \sum_{m=1}^{M} [f_0^i]_n \left(\sum_{m=1}^{m} a_m(t)\boldsymbol{\phi}_m(\mathbf{z}_m^{[f_0^i]_n}) \right) \psi_m^{[f_0^i]_n}(\mathbf{x}), \qquad n = 1, ..., M \tag{121}$$

$$f_j^i \approx f_j^i(\mathbf{a}_M) = \sum_{m=1}^{M} f_j^i \left(\sum_{m=1}^{m} a_m(t)\boldsymbol{\phi}_m(\mathbf{z}_m^{f_j^i}) \right) \psi_m^{f_j^i}(\mathbf{x}), \qquad j = 1, ..., 9 \tag{122}$$

It is convenient to write (120) in matrix/vector form, as would be required for numerical implementation. To do this, note that, for the i^{th}, $i = 1, ..., 5$ component of $\mathbf{f}_j(\mathbf{a}_M)$ and for $l = 1, ..., M$, we have that (implied summation on the $i = 1, ..., 5$, and letting ϕ_l^i denote the i^{th} component of $\boldsymbol{\phi}_l$):

$$\begin{aligned}
(\boldsymbol{\phi}_l, \mathbf{f}_j(\mathbf{a}_M)) &= \left(\phi_l^i, f_j^i(\mathbf{a}_M)\right) \\
&= \left(\phi_l^i, \textstyle\sum_{m=1}^{M} f_j^i \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^i}) \right) \psi_m^{f_j^i} \right) \\
&= \textstyle\int_\Omega \phi_l^i \left[\sum_{m=1}^{M} \left\{ f_j^i \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^i}) \right) \right\} \psi_m^{f_j^i} \right] d\Omega \\
&= \underbrace{\sum_{m=1}^{M} \left[\int_\Omega (\phi_l^1 \psi_m^{f_j^1}, \phi_l^2 \psi_m^{f_j^2}, \phi_l^3 \psi_m^{f_j^3}, \phi_l^4 \psi_m^{f_j^4}, \phi_l^5 \psi_m^{f_j^5}) d\Omega \right]}_{\in \mathbb{R}^{1\times 5}} \underbrace{\mathbf{f}_j \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{\mathbf{f}_j}) \right)}_{\in \mathbb{R}^{5\times 1}}
\end{aligned} \tag{123}$$

where, to further clarify the notation:

$$\mathbf{f}_j \left(\sum_{n=1}^{N} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{\mathbf{f}_j}) \right) \equiv \begin{pmatrix} f_j^1 \left(\sum_{n=1}^{m} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^1}) \right) \\ f_j^2 \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^2}) \right) \\ f_j^3 \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^3}) \right) \\ f_j^4 \left(\sum_{n=1}^{M} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^4}) \right) \\ f_j^5 \left(\sum_{n=1}^{m} a_n \boldsymbol{\phi}_n(\mathbf{z}_m^{f_j^5}) \right) \end{pmatrix} \in \mathbb{R}^5 \tag{124}$$

(123) is a matrix/vector product of the form $\mathbf{G}^{\mathbf{f}_j}\mathbf{f}_j(\mathbf{D}^{\mathbf{f}_j}\mathbf{a}_M)$ where

$$\mathbf{G}^{\mathbf{f}_j}_{l,[5(m-1)+1:5m]} = \int_\Omega (\phi_l^1\psi_m^{f_j^1}, \phi_l^2\psi_m^{f_j^2}, \phi_l^3\psi_m^{f_j^3}, \phi_l^4\psi_m^{f_j^4}, \phi_l^5\psi_m^{f_j^5})d\Omega \in \mathbb{R}^{1\times 5} \tag{125}$$

for $1 \le l,m \le M$ (so that $\mathbf{G}^{\mathbf{f}_j} \in \mathbb{R}^{M\times 5M}$), and

$$\mathbf{D}^{\mathbf{f}_j} \equiv \begin{pmatrix} \boldsymbol{\phi}_1(\mathbf{z}_1^{\mathbf{f}_j}) & \dots & \boldsymbol{\phi}_N(\mathbf{z}_1^{\mathbf{f}_j}) \\ \vdots & \ddots & \vdots \\ \boldsymbol{\phi}_1(\mathbf{z}_M^{\mathbf{f}_j}) & \dots & \boldsymbol{\phi}_N(\mathbf{z}_M^{\mathbf{f}_j}) \end{pmatrix} \in \mathbb{R}^{5M\times N} \tag{126}$$

By $\boldsymbol{\phi}_m(\mathbf{z}_n^{\mathbf{f}_j})$ we mean

$$\boldsymbol{\phi}_m(\mathbf{z}_n^{\mathbf{f}_j}) \equiv \begin{pmatrix} \phi_m^1(\mathbf{z}_n^{f_j^1}) \\ \phi_m^2(\mathbf{z}_n^{f_j^2}) \\ \phi_m^3(\mathbf{z}_n^{f_j^3}) \\ \phi_m^4(\mathbf{z}_n^{f_j^4}) \\ \phi_m^5(\mathbf{z}_n^{f_j^5}) \end{pmatrix} \in \mathbb{R}^5, \qquad 1 \le m,n \le M \tag{127}$$

$\mathbf{f}_j(\mathbf{D}^{\mathbf{f}_j}\mathbf{a}_M) \in \mathbb{R}^{5M}$ is defined analogously to (38).

Similarly, turning one's attention to the left-hand side of (120) (with implied summation on $i = 1,\dots,5$ as in (123)):

$$\begin{aligned}
\textstyle\sum_{k=1}^M(\boldsymbol{\phi}_l, [\mathbf{f}_0(\mathbf{a}_M)]_k)\dot{a}_k &= \textstyle\sum_{k=1}^M \left(\phi_l^i, [f_0^i(\mathbf{a}_M)]_k\right)\dot{a}_k \\
&= \textstyle\sum_{k=1}^M \left(\phi_l^i, \sum_{m=1}^M \left[f_0^i\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n(\mathbf{z}_m^{[f_0^i]_k})\right)\right]_k \psi_m^{[f_0^i]_k}\right)\dot{a}_k \\
&= \textstyle\sum_{k=1}^M \dot{a}_k \int_\Omega \phi_l^i \left\{\sum_{m=1}^M \left[f_0^i\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n(\mathbf{z}_m^{[f_0^i]_k})\right)\right]_k \psi_m^{[f_0^i]_k}\right\} d\Omega \\
&= \textstyle\sum_{k=1}^M \left\{ \sum_{m=1}^M \underbrace{\left[\int_\Omega (\phi_l^1\psi_m^{[f_0^1]_k}, \phi_l^2\psi_m^{[f_0^2]_k}, \phi_l^3\psi_m^{[f_0^3]_k}, \phi_l^4\psi_m^{[f_0^4]_k}, \phi_l^5\psi_m^{[f_0^5]_k})d\Omega\right]}_{\in\mathbb{R}^{1\times 5}} \underbrace{\left[\mathbf{f}_0\left(\sum_{n=1}^M a_n\boldsymbol{\phi}_n(\mathbf{z}_m^{[\mathbf{f}_0]_k})\right)\right]_k}_{\in\mathbb{R}^{5\times 1}} \right\}\dot{a}_k
\end{aligned} \tag{128}$$

The entries of the mass matrix can be "read off" from (128), namely

$$\mathbf{M}_{[1:M],k} = \mathbf{G}^{[\mathbf{f}_0]_k}[\mathbf{f}_0]_k\left(\mathbf{D}^{[\mathbf{f}_0]_k}\mathbf{a}_M\right) \in \mathbb{R}^M \tag{129}$$

for $1 \le k \le M$, where $\mathbf{G}^{[\mathbf{f}_0]_k}$ and $\mathbf{D}^{[\mathbf{f}_0]_k}$ are defined analogously to (125) and (126) respectively.

With this notation in place, (120) can be written in matrix/vector form as

$$\boxed{\mathbf{M}\dot{\mathbf{a}}_M = \mathbf{G}^{\mathbf{f}_i}\mathbf{f}_i(\mathbf{D}^{\mathbf{f}_i}\mathbf{a}_M) + \mathbf{G}^4\mathbf{f}_4(\mathbf{D}^{\mathbf{f}_4}\mathbf{a}_M) - \mathbf{G}^{\mathbf{f}_{i+4}}\mathbf{f}_{i+4}(\mathbf{D}^{\mathbf{f}_{i+4}}\mathbf{a}_M) - \mathbf{G}^{\mathbf{f}_8}\mathbf{f}_8(\mathbf{D}^{\mathbf{f}_8}\mathbf{a}_M) - \mathbf{G}^{\mathbf{f}_9}\mathbf{f}_9(\mathbf{D}^{\mathbf{f}_9}\mathbf{a}_M)} \tag{130}$$

(implied summation on $i = 1,2,3$). (130) can be integrated in time using a standard explicit time-integration scheme, or an implicit time-integration scheme, with the application of Newton's method at each time step. We emphasize again that the upshot of formulating the ROM *with* interpolation is all the inner-products are contained in the $\mathbf{G}^{\mathbf{f}_j}$ matrices (125), which can be pre-computed prior to time integration of and/or application of Newton's method to the ROM ODE system (130). Similarly, the interpolated mass matrix (129) can also be pre-computed. The time-integration of the ROM ODE system (130) will require inversion of this matrix, but since the number of modes M will in general be quite small, the relative cost of this inversion is miniscule.

4 Conclusions and Future Work

The present work has focused on techniques for building entropy-stable and reduced order models (ROMs) governed by non-linear partial differential equations (PDEs) in fluid mechanics. It turns out that one can bypass the need to

recompute inner products involving the non-linear terms at each time or Newton step, thereby reducing the on-line computational complexity of the ROM, by handling the nonlinearities using a "best" points interpolation algorithm [15, 16].

The said "best" points interpolation approach was tested on a model one-dimensional (1D) convection-diffusion-reaction system of equations representing the flow through a non-adiabatic tubular reactor [7]. Numerical tests on this simple non-linear problem revealed that the interpolation procedure successfully captures the non-linear behavior (e.g., limit cycles) of the solution when a spectral basis is employed. It also revealed some shortcomings of the Proper Orthogonal Decomposition (POD) basis that one may wish to examine further in future work.

Following this preliminary testing of the interpolation, attention was turned to the key equations in fluid dynamics, namely the compressible three-dimensional (3D) Navier-Stokes equations. The nonlinearity present in these equations presents a challenge for developing provably stable ROMs. This challenge was addressed with the help of a transformation that effectively symmetrizes these equations, leading to a projection technique that leads to a model that obeys the second law of thermodynamics: non-decreasing entropy of the solution. Following a proof of the entropy-stability of the ROM solution with appropriate boundary condition, an efficient "best" points interpolation procedure was formulated to handle the non-linear terms in the symmetrized equations. Given this formulation, it should be straight forward to implement the Navier-Stokes ROM with the proposed interpolation, and to test the performance of this solution under different choices of bases. Future work will involve implementation and testing of the Navier-Stokes reduced order model formulated herein.

5 Appendix

5.1 Euler Fluxes in the Entropy Variables and Symmetrized Matrices

To simplify the notation, let us introduce the following variables[9]:

$$\begin{array}{lll} \bar{\gamma}=\gamma-1, & k_1=\frac{1}{2V_5}(V_2^2+V_3^2+V_4^2), & k_2=k_1-\gamma, \\ k_3=k_1^2-2\gamma k_1+\gamma, & k_4=k_2-\bar{\gamma}, & k_5=k_2^2-\bar{\gamma}(k_1+k_2), \\ c_1=\bar{\gamma}V_5-V_2^2, & d_1=-V_2V_3, & e_1=V_2V_5, \\ c_2=\bar{\gamma}V_5-V_3^2, & d_2=-V_2V_4, & e_2=V_3V_5, \\ c_3=\bar{\gamma}V_5-V_4^2, & d_3=-V_3V_4, & e_3=V_4V_5. \end{array} \tag{131}$$

In the entropy variables $\mathbf{V}$, the Euler fluxes $\mathbf{F}_i(\mathbf{V})$ are given by:

$$\mathbf{F}_1(\mathbf{V})=\frac{\rho_1}{V_5}\begin{pmatrix} e_1 \\ c_1 \\ d_1 \\ d_2 \\ k_2V_2 \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{V})=\frac{\rho_1}{V_5}\begin{pmatrix} e_2 \\ d_1 \\ c_2 \\ d_3 \\ k_2V_3 \end{pmatrix}, \quad \mathbf{F}_3(\mathbf{V})=\frac{\rho_1}{V_5}\begin{pmatrix} e_3 \\ d_2 \\ d_3 \\ c_3 \\ k_2V_4 \end{pmatrix} \tag{132}$$

The symmetrizing matrix $\mathbf{A}_0$ and its inverse are given by

$$\mathbf{A}_0=\mathbf{U}_{,\mathbf{V}}=\frac{\rho_1}{\bar{\gamma}V_5}\begin{pmatrix} -V_5^2 & e_1 & e_2 & e_3 & V_5(1-k_1) \\ & c_1 & d_1 & d_2 & V_2k_2 \\ & & c_2 & d_3 & V_3k_2 \\ & & & c_3 & V_4k_2 \\ \text{symm.} & & & & -k_3 \end{pmatrix} \tag{133}$$

[9] This section is repeated here from the Appendix of [9] to make this document self-contained.

and

$$\mathbf{A}_0^{-1} = \mathbf{V}_{,\mathbf{U}} = -\frac{1}{\rho_1 V_5}\begin{pmatrix} k_1^2+\gamma & k_1V_2 & k_1V_3 & k_1V_4 & (k_1+1)V_5 \\ & V_2^2-V_5 & -d_1 & -d_2 & e_1 \\ & & V_3^2-V_5 & -d_3 & e_2 \\ & & & V_4^2-V_5 & e_3 \\ \text{symm.} & & & & V_5^2 \end{pmatrix} \tag{134}$$

The Jacobians of the Euler fluxes are:

$$\tilde{\mathbf{A}}_1 = \mathbf{F}_{1,\mathbf{V}} = \frac{\rho_1}{\bar{\gamma}V_5^2}\begin{pmatrix} e_1V_5 & c_1V_5 & d_1V_5 & d_2V_5 & k_2e_1 \\ & -(c_1+2\bar{\gamma}V_5)V_2 & -c_1V_3 & -c_1V_4 & c_1k_2+\bar{\gamma}V_2^2 \\ & & -c_2V_2 & -d_1V_4 & k_4d_1 \\ & & & -c_3V_2 & k_4d_2 \\ \text{symm.} & & & & k_5V_2 \end{pmatrix} \tag{135}$$

$$\tilde{\mathbf{A}}_2 = \mathbf{F}_{2,\mathbf{V}} = \frac{\rho_1}{\bar{\gamma}V_5^2}\begin{pmatrix} e_2V_5 & d_1V_5 & c_2V_5 & d_3V_5 & k_2e_2 \\ & -c_1V_3 & -c_2V_2 & -d_1V_4 & k_4d_1 \\ & & -(c_2+2\bar{\gamma}V_5)V_3 & -c_2V_4 & c_2k_2+\bar{\gamma}V_3^2 \\ & & & -c_3V_3 & k_4d_3 \\ \text{symm.} & & & & k_5V_3 \end{pmatrix} \tag{136}$$

$$\tilde{\mathbf{A}}_3 = \mathbf{F}_{3,\mathbf{V}} = \frac{\rho_1}{\bar{\gamma}V_5^2}\begin{pmatrix} e_3V_5 & d_2V_57d_3V_5 & c_3V_5 & k_2e_3 & \\ & -c_1V_4 & -d_2V_3 & -c_3V_2 & k_4d_2 \\ & & -c_2V_4 & -c_3V_3 & k_4d_3 \\ & & & -(c_3+2\bar{\gamma}V_5)V_4 & c_3k_2+\bar{\gamma}V_4^2 \\ \text{symm.} & & & & k_5V_4 \end{pmatrix} \tag{137}$$

The velocity and temperature can be written in the entropy variables as:

$$u_i(\mathbf{V}) = -\frac{V_{i+1}}{V_5}, \;\; i = 1,2,3 \tag{138}$$

$$\theta(\mathbf{V}) = -\frac{1}{c_v V_5} \tag{139}$$

The gradients of the viscous and heat fluxes are given by:

$$u_{i,j} = \frac{-V_5V_{i+1,j}+V_{i+1}V_{5,j}}{V_5^2} \tag{140}$$

$$\kappa\theta_{,i} = \frac{\gamma\mu}{Pr}\frac{1}{V_5^2}V_{5,i} \tag{141}$$

where $Pr \equiv \mu c_p/\kappa$ is the Prandtl number.

Finally, the symmetrized viscous and heat flux matrices $\tilde{\mathbf{K}}_{ij} \equiv \tilde{\mathbf{K}}_{ij}^v + \tilde{\mathbf{K}}_{ij}^h$ are given by:

$$\tilde{\mathbf{K}}_{11} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -(\gamma-2\mu)V_5^2 & 0 & 0 & (\lambda+2\mu)e_1 \\ 0 & 0 & -\mu V_5^2 & 0 & \mu e_2 \\ 0 & 0 & 0 & -\mu V_5^2 & \mu e_3 \\ 0 & (\lambda+2\mu)e_1 & \mu e_2 & \mu e_3 & -\left[(\lambda+2\mu)V_2^2+\mu(V_3^2+V_4^2)-\frac{\gamma\mu V_5}{Pr}\right] \end{pmatrix} \tag{142}$$

$$\tilde{\mathbf{K}}_{12} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\lambda V_5^2 & 0 & \lambda e_2 \\ 0 & -\mu V_5^2 & 0 & 0 & \mu e_1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \mu e_2 & \lambda e_1 & 0 & (\lambda+\mu)d_1 \end{pmatrix} \tag{143}$$

$$\tilde{\mathbf{K}}_{13} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\lambda V_5^2 & \lambda e_3 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -\mu V_5^2 & 0 & 0 & \mu e_1 \\ 0 & \mu e_3 & 0 & \lambda e_1 & (\lambda+\mu) d_2 \end{pmatrix} \tag{144}$$

$$\tilde{\mathbf{K}}_{22} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\mu V_5^2 & 0 & 0 & \mu e_1 \\ 0 & 0 & -(\lambda+2\mu)V_5^2 & 0 & (\lambda+2\mu)e_2 \\ 0 & 0 & 0 & -\mu V_5^2 & \mu e_3 \\ 0 & \mu e_1 & (\lambda+2\mu)e_2 & \mu e_3 & -\left[(\lambda+2\mu)V_3^2 + \mu(V_2^2+V_4^2) - \frac{\gamma\mu V_5}{Pr}\right] \end{pmatrix} \tag{145}$$

$$\tilde{\mathbf{K}}_{23} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\lambda V_5^2 & \lambda e_3 \\ 0 & 0 & -\mu V_5^2 & 0 & \mu e_2 \\ 0 & 0 & \mu e_3 & \lambda e_2 & (\lambda+\mu) d_3 \end{pmatrix} \tag{146}$$

$$\tilde{\mathbf{K}}_{33} = \frac{1}{V_5^3}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -\mu V_5^2 & 0 & 0 & \mu e_1 \\ 0 & 0 & -\mu V_5^2 & 0 & \mu e_2 \\ 0 & 0 & 0-(\lambda+2\mu)V_5^2 & (\lambda+2\mu)e_3 & \\ 0 & \mu e_1 & \mu e_2 & (\lambda+2\mu)e_3 & -\left[(\lambda+2\mu)V_4^2 + \mu(V_2^2+V_3^2) - \frac{\gamma\mu V_5}{Pr}\right] \end{pmatrix} \tag{147}$$

with

$$\tilde{\mathbf{K}}_{21} = \tilde{\mathbf{K}}_{12}^T, \quad \tilde{\mathbf{K}}_{31} = \tilde{\mathbf{K}}_{13}^T, \quad \tilde{\mathbf{K}}_{32} = \tilde{\mathbf{K}}_{23}^T \tag{148}$$

5.2 The Matrix $\tilde{\mathbf{A}}_i n_i$ and Application of No-Penetration Boundary Condition

Given the symmetrized Euler flux matrices (135)–(137), and letting $\mathbf{n}^T = (n_1, n_2, n_3)$ denote an outward normal vector to some boundary $\partial\Omega_W$ in the domain, one has that:

$$\tilde{\mathbf{A}}_i n_i = \begin{pmatrix} \frac{\rho}{\gamma-1}(\mathbf{u}\cdot\mathbf{n}) & \rho\imath n_1 + \frac{\rho}{\gamma-1}(\mathbf{u}\cdot\mathbf{n})u_1 & \rho\imath n_2 + \frac{\rho}{\gamma-1}(\mathbf{u}\cdot\mathbf{n})u_2 \\ & \rho\imath(\mathbf{u}\cdot\mathbf{n}) + 2\rho\imath u_1 n_1 + \frac{\rho}{\gamma-1}u_1^2(\mathbf{u}\cdot\mathbf{n}) & \rho\imath(u_2 n_1 + u_1 n_2) + \frac{\rho}{\gamma-1}u_1 u_2(\mathbf{u}\cdot\mathbf{n}) \\ & & \rho\imath(\mathbf{u}\cdot\mathbf{n}) + 2\rho\imath u_2 n_2 + \frac{\rho}{\gamma-1}u_2^2(\mathbf{u}\cdot\mathbf{n}) \\ & \text{symm.} & \end{pmatrix}$$

$$\begin{pmatrix} \rho\imath n_3 + \frac{\rho}{\gamma-1}(\mathbf{u}\cdot\mathbf{n})u_3 & \frac{\rho}{\gamma-1}(\mathbf{u}\cdot\mathbf{n})\left[\frac{1}{2}u^2 + \gamma\imath\right] \\ \rho\imath[u_3 n_1 + u_1 n_3] + \frac{\rho}{\gamma-1}u_1 u_3(\mathbf{u}\cdot\mathbf{n}) & \rho\imath\left(\frac{1}{2}u^2 + \gamma\imath\right)n_1 + \frac{\rho}{\gamma-1}\left[\left(\frac{1}{2}u^2 + \imath\gamma\right) + \imath(\gamma-1)\right]u_1(\mathbf{u}\cdot\mathbf{n}) \\ \frac{\rho}{\gamma-1}u_2 u_3(\mathbf{u}\cdot\mathbf{n}) + \rho\imath(u_3 n_2 + u_2 n_3) & \rho\imath\left(\frac{1}{2}u^2 + \imath\gamma\right)n_2 + \frac{\rho}{\gamma-1}\left[\left(\frac{1}{2}u^2 + \imath\gamma\right) + \imath(\gamma-1)\right]u_2(\mathbf{u}\cdot\mathbf{n}) \\ \rho\imath(\mathbf{u}\cdot\mathbf{n}) + 2\rho\imath u_3 n_3 + \frac{\rho}{\gamma-1}u_3^2(\mathbf{u}\cdot\mathbf{n}) & \left(\frac{1}{2}u^2 + i\gamma\right)\rho\imath n_3 + \frac{\rho}{\gamma-1}\left(\left[\frac{1}{2}u^2 + \imath\gamma\right] + \imath[\gamma-1]\right)u_3(\mathbf{u}\cdot\mathbf{n}) \\ & \frac{\rho}{\gamma-1}\left[\frac{1}{4}u^4 + \imath(u^2+\gamma\imath)(2\gamma-1)\right](\mathbf{u}\cdot\mathbf{n}) \end{pmatrix} \tag{149}$$

It is straight forward to apply the no-penetration boundary condition, $\mathbf{u}\cdot\mathbf{n} = 0$ on $\partial\Omega_W$ to (149):

$$[\tilde{\mathbf{A}}_i n_i]^{np} = \begin{pmatrix} 0 & \rho\imath n_1 & \rho\imath n_2 & \rho\imath n_3 & 0 \\ & 2\rho\imath u_1 n_1 & \rho\imath(u_2 n_1 + u_1 n_2) & \rho\imath[u_3 n_1 + u_1 n_3] & \rho\imath\left(\frac{1}{2}u^2 + \gamma\imath\right)n_1 \\ & & 2\rho\imath u_2 n_2 & \rho\imath(u_3 n_2 + u_2 n_3) & \rho\imath\left(\frac{1}{2}u^2 + \imath\gamma\right)n_2 \\ & & & 2\rho\imath u_3 n_3 & \left(\frac{1}{2}u^2 + \imath\gamma\right)\rho\imath n_3 \\ & & & & 0 \end{pmatrix} \tag{150}$$

so that

$$[\tilde{\mathbf{A}}_i n_i \mathbf{V}]^{np} = \begin{pmatrix} 0 \\ -\rho \imath s n_1 \\ -\rho \imath s n_2 \\ -\rho \imath s n_3 \\ 0 \end{pmatrix} \tag{151}$$

and

$$\begin{aligned} \mathbf{V}^T [\mathbf{A}_i n_i \mathbf{V}]^{np} &= -\rho s(\mathbf{u} \cdot \mathbf{n}) \\ &= H u_i n_i \\ &= \sigma_i n_i \end{aligned} \tag{152}$$

Note that by the divergence theorem,

$$\int_\Omega (Hu_i), i d\Omega = \int_\Omega \sigma_{i,i} d\Omega = \int_{\partial\Omega_W} \sigma_i n_i dS \tag{153}$$

5.3 The Matrices $\tilde{\mathbf{K}}_{ij}^{\nu} n_i$ and Application of No-Slip Boundary Condition

From (142)–(148), letting $\mathbf{n}^T = (n_1, n_2, n_3)$ denote the outward unit normal vector to some relevant boundary $\partial\Omega_W$:

$$\tilde{\mathbf{K}}_{11}^{\nu} \mathbf{V}_{,1} = \begin{pmatrix} 0 \\ (\lambda + 2\mu) u_{1,1} \\ \mu u_{2,1} \\ \mu u_{3,1} \\ -(\lambda + 2\mu) \frac{V_2}{V_5} u_{1,1} - \mu \frac{V_3}{V_5} u_{2,1} - \mu \frac{V_4}{V_5} u_{3,1} \end{pmatrix} \tag{154}$$

$$\tilde{\mathbf{K}}_{12}^{\nu} \mathbf{V}_{,2} = \begin{pmatrix} 0 \\ \lambda u_{2,2} \\ \mu u_{1,2} \\ 0 \\ -\mu \frac{V_3}{V_5} u_{1,2} - \lambda \frac{V_2}{V_5} u_{2,2} \end{pmatrix} \tag{155}$$

$$\tilde{\mathbf{K}}_{13}^{\nu} \mathbf{V}_{,3} = \begin{pmatrix} 0 \\ \lambda u_{3,3} \\ 0 \\ \mu u_{1,3} \\ -\mu \frac{V_4}{V_5} u_{1,3} - \lambda \frac{V_2}{V_5} u_{3,3} \end{pmatrix} \tag{156}$$

So that

$$\tilde{\mathbf{K}}_{1j} \mathbf{V}_{,j} = \begin{pmatrix} 0 \\ (\lambda + 2\mu) u_{1,1} + \lambda u_{2,2} + \lambda u_{3,3} \\ \mu u_{2,1} + \mu u_{1,2} \\ \mu u_{3,1} + \mu u_{1,3} \\ -(\lambda + 2\mu) \frac{V_2}{V_5} u_{1,1} - \mu \frac{V_3}{V_5} u_{2,1} - \mu \frac{V_4}{V_5} u_{3,1} - \mu \frac{V_3}{V_5} u_{1,2} - \lambda \frac{V_2}{V_5} u_{2,2} - \mu \frac{V_4}{V_5} u_{1,3} - \lambda \frac{V_2}{V_5} u_{3,3} \end{pmatrix} \tag{157}$$

meaning

$$\begin{aligned} \mathbf{V}^T \tilde{\mathbf{K}}_{1j} \mathbf{V}_{,j} &= (\lambda + 2\mu) u_{1,1} V_2 + \lambda u_{2,2} V_2 + \lambda u_{3,3} V_2 + \mu u_{2,1} V_3 + \mu u_{1,2} V_3 + \mu u_{3,1} V_4 + \mu u_{1,3} V_4 \\ &\quad -(\lambda + 2\mu) V_2 u_{1,1} - \mu V_3 u_{2,1} - \mu V_4 u_{3,1} - \mu V_3 u_{1,2} - \lambda V_2 u_{2,2} - \mu V_4 u_{1,3} - \lambda V_2 u_{3,3} \\ &= 0 \end{aligned} \tag{158}$$

(as asserted by Hughes in [9]). Moreover, applying the no-slip condition $\mathbf{u} = \mathbf{0}$ on $\partial\Omega_W$, one has that

$$[\tilde{\mathbf{K}}_{1j} \mathbf{V}_{,j}]^{ns} = \begin{pmatrix} 0 \\ 2\mu u_{1,1} + \lambda (u_{1,1} + u_{2,2} + u_{3,3}) \\ \mu (u_{2,1} + u_{1,2}) \\ \mu (u_{3,1} + u_{1,3}) \\ 0 \end{pmatrix} \tag{159}$$

Next:

$$\tilde{\mathbf{K}}^{v}_{21}\mathbf{V}_{,1} = \frac{1}{V_5^3}\begin{pmatrix} 0 \\ \mu V_5[-V_5V_{3,1}+\mu V_3V_{5,1}] \\ \lambda V_5[-V_5V_{2,1}+V_2V_{5,1}] \\ 0 \\ -\lambda V_3[-V_5V_{2,1}+V_2V_{5,1}]+\mu V_2[V_5V_{3,1}-V_3V_{5,1}] \end{pmatrix} \tag{160}$$

$$\tilde{\mathbf{K}}^{v}_{22}\mathbf{V}_{,2} = \begin{pmatrix} 0 \\ \mu u_{1,2} \\ (\lambda+2\mu)u_{2,2} \\ \mu u_{3,2} \\ -\mu\frac{V_2}{V_5}u_{1,2}-\mu\frac{V_4}{V_5}u_{3,2}-(\lambda+2\mu)\frac{V_3}{V_5}u_{2,2} \end{pmatrix} \tag{161}$$

$$\tilde{\mathbf{K}}^{v}_{23}\mathbf{V}_{,3} = \begin{pmatrix} 0 \\ 0 \\ \lambda u_{3,3} \\ \mu u_{2,3} \\ -\mu\frac{V_4}{V_5}u_{2,3}-\lambda\frac{V_3}{V_5}u_{3,3} \end{pmatrix} \tag{162}$$

It follows that

$$\mathbf{K}_{2j}\mathbf{V}_{,j} = \begin{pmatrix} 0 \\ \mu u_{2,1}+\mu u_{1,2} \\ \lambda u_{1,1}+(\lambda+2\mu)u_{2,2}+\lambda u_{3,3} \\ \mu u_{3,2}+\mu u_{2,3} \\ -\lambda\frac{V_3}{V_5}u_{1,1}-\mu\frac{V_2}{V_5}u_{2,1}-\mu\frac{V_4}{V_5}u_{2,3}-\lambda\frac{V_3}{V_5}u_{3,3}-\mu\frac{V_2}{V_5}u_{1,2}-\mu\frac{V_4}{V_5}u_{3,2}-(\lambda+2\mu)\frac{V_3}{V_5}u_{2,2} \end{pmatrix} \tag{163}$$

so that

$$\begin{aligned} \mathbf{V}^T\mathbf{K}_{2j}\mathbf{V}_{,j} \quad &= \mu u_{2,1}V_2+\mu u_{1,2}V_2+\lambda u_{1,1}V_3+(\lambda+2\mu)u_{2,2}V_3+\lambda u_{3,3}V_3+\mu u_{3,2}V_4+\mu u_{2,3}V_4 \\ &-\lambda V_3u_{1,1}-\mu V_2u_{2,1}-\mu V_4u_{2,3}-\lambda V_3u_{3,3}-\mu V_2u_{1,2}-\mu V_4u_{3,2}-(\lambda+2\mu)V_3u_{2,2} \\ &= 0 \end{aligned} \tag{164}$$

(also as Hughes asserts [9]). Then, applying the no-slip condition, $\mathbf{u}=\mathbf{0}$ on $\partial\Omega_W$:

$$[\mathbf{K}_{2j}\mathbf{V}_{,j}]^{ns} = \begin{pmatrix} 0 \\ \mu(u_{2,1}+u_{1,2}) \\ \lambda(u_{1,1}+u_{2,2}+u_{3,3})+2\mu u_{2,2} \\ \mu(u_{3,2}+u_{2,3}) \\ 0 \end{pmatrix} \tag{165}$$

Finally:

$$\tilde{\mathbf{K}}^{v}_{31}\mathbf{V}_{,1} = \frac{1}{V_5^3}\begin{pmatrix} 0 \\ \mu u_{3,1} \\ 0 \\ \lambda u_{1,1} \\ -\lambda\frac{V_4}{V_5}u_{1,1}-\mu\frac{V_2}{V_5}u_{3,1} \end{pmatrix} \tag{166}$$

$$\tilde{\mathbf{K}}^{v}_{32}\mathbf{V}_{,2} = \begin{pmatrix} 0 \\ 0 \\ \mu u_{3,2} \\ \lambda u_{2,2} \\ -\lambda\frac{V_4}{V_5}u_{2,2}-\mu\frac{V_3}{V_5}u_{3,2} \end{pmatrix} \tag{167}$$

$$\tilde{\mathbf{K}}^{v}_{33}\mathbf{V}_{,3} = \begin{pmatrix} 0 \\ \mu u_{1,3} \\ \mu u_{2,3} \\ (\lambda+2\mu)u_{3,3} \\ -\mu\frac{V_2}{V_5}u_{1,3}-\mu\frac{V_3}{V_5}u_{2,3}-(\lambda+2\mu)\frac{V_4}{V_5}u_{3,3} \end{pmatrix} \tag{168}$$

so that

$$\tilde{\mathbf{K}}^{v}_{3j}\mathbf{V}_{,j} = \begin{pmatrix} 0 \\ \mu(u_{3,1}+u_{1,3}) \\ \mu(u_{3,2}+u_{2,3}) \\ \lambda(u_{1,1}+u_{2,2}+u_{3,3})+2\mu u_{3,3} \\ -\lambda\frac{V_4}{V_5}u_{1,1}-\mu\frac{V_2}{V_5}u_{3,1}-\lambda\frac{V_4}{V_5}u_{2,2}-\mu\frac{V_3}{V_5}u_{3,2}-\mu\frac{V_2}{V_5}u_{1,3}-\mu\frac{V_3}{V_5}u_{2,3}-(\lambda+2\mu)\frac{V_4}{V_5}u_{3,3} \end{pmatrix} \tag{169}$$

which confirms that

$$\begin{aligned} \mathbf{V}^T\tilde{\mathbf{K}}^{v}_{3j}\mathbf{V}_{,j} &= \mu u_{3,1}V_2+\mu u_{1,3}V_2+\mu u_{3,2}V_3+\mu u_{2,3}V_3+\lambda u_{1,1}V_4+\lambda u_{2,2}V_4+(\lambda+2\mu)u_{3,3}V_4 \\ &\quad -\lambda V_4u_{1,1}-\mu V_2u_{3,1}-\lambda V_4u_{2,2}-\mu V_3u_{3,2}-\mu V_2u_{1,3}-\mu V_3u_{2,3}-(\lambda+2\mu)V_4u_{3,3} \\ &= 0 \end{aligned} \tag{170}$$

As for the application of the no-slip condition ($\mathbf{u}=\mathbf{0}$ on $\partial\Omega_W$):

$$[\tilde{\mathbf{K}}^{v}_{3j}\mathbf{V}_{,j}]^{ns} = \begin{pmatrix} 0 \\ \mu(u_{3,1}+u_{1,3}) \\ \mu(u_{3,2}+u_{2,3}) \\ \lambda(u_{1,1}+u_{2,2}+u_{3,3})+2\mu u_{3,3} \\ 0 \end{pmatrix} \tag{171}$$

Putting everything together, we obtain the matrix stemming from the application of the no-slip condition $\mathbf{u}=\mathbf{0}$ on $\partial\Omega_W$:

$$[\tilde{\mathbf{K}}_{ij}n_i\mathbf{V}_{,j}]^{ns} = \frac{\mu}{V_5^2}\begin{pmatrix} 0 \\ (-V_5V_{i+1,1}+V_{i+1}V_{5,1}-V_5V_{2,i}+V_2V_{5,i})n_i \\ (-V_5V_{i+1,2}+V_{i+1}V_{5,2}-V_5V_{3,i}+V_3V_{5,i})n_i \\ (-V_5V_{i+1,3}+V_{i+1}V_{5,3}-V_5V_{4,i}+V_4V_{5,i})n_i \\ 0 \end{pmatrix} + \lambda\left[\frac{-V_5V_{i+1,i}+V_{i+1}V_{5,i}}{V_5^2}\right]\begin{pmatrix} 0 \\ n_1 \\ n_2 \\ n_3 \\ 0 \end{pmatrix} \tag{172}$$

An interesting observation is that components $(2:4)$ of $[\tilde{\mathbf{K}}_{ij}n_i\mathbf{V}_{,j}]^{ns}$ are

$$[\tilde{\mathbf{K}}_{ij}n_i\mathbf{V}_{,j}]^{ns}_{2:4} = [2\mu\mathbf{S}+\lambda\nabla\cdot\mathbf{u}\mathbf{I}]\mathbf{n} \tag{173}$$

where $\mathbf{S}$ is the strain tensor, with components given by

$$S_{ij} = \frac{1}{2}(u_{i,j}+u_{j,i}) \tag{174}$$

Recall that the general deformation law for a Newtonian viscous fluid is (see equation (11) in [9]):

$$\tau_{ij} = 2\mu S_{ij}+\delta_{ij}\lambda\nabla\cdot\mathbf{u} \tag{175}$$

Moreover, the governing momentum equations have the form:

$$\rho\frac{D\mathbf{u}}{Dt} = \nabla\cdot\tau_{ij} \tag{176}$$

If we set $\frac{D\mathbf{u}}{Dt}=\mathbf{0}$ at the wall (i.e., assume the fluid is at rest at the wall), then (176) implies that $\nabla\cdot\tau_{ij}=0$ at the wall, or $\tau_{ij}\cdot\mathbf{n}=0$ at the wall (by the divergence theorem). Then (175) implies that $[2\mu\mathbf{S}+\lambda\nabla\cdot\mathbf{u}\mathbf{I}]\mathbf{n}=\mathbf{0}$ at the wall.

5.4 The Matrices $\tilde{\mathbf{K}}^{h}_{ij}n_i$ and Application of the Adiabatic-Wall Boundary Condition

Let θ denote the absolute temperature. Then, from (45) in [9],

$$\tilde{\mathbf{K}}^{h}_{ij}n_i\mathbf{V}_{,j} = \mathbf{F}^{h}_{i}n_i = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \kappa\theta_{,i}n_i \end{pmatrix} \tag{177}$$

Suppose the wall is adiabatic, i.e., $\theta_{,i}n_i = 0$. Then

$$[\tilde{\mathbf{K}}^h_{ij} n_i \mathbf{V}_{,j}]^{ad} = 0 \tag{178}$$

5.5 Proof of Indefiniteness of Heat Flux Matrix $\tilde{\mathbf{K}}^h_{ij}$ with Harten's Family of Homogeneous Generalized Entropy Flux Functions

Suppose we wish to use a homogeneous entropy flux function, from Harten's family of homogeneous generalized entropy flux functions [6]:

$$H(s) = Ke^{\frac{s}{\alpha+\gamma}} \tag{179}$$

where $\alpha, K \in \mathbb{R}$.

Without loss of generality, consider the 1D case. Refer to [5]. In particular, we have:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} \equiv \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} \tag{180}$$

where

$$p = (\gamma - 1)\left(E - \frac{1}{2}\rho u^2\right) \tag{181}$$

It is shown in [5] that with the generalized entropy flux function (179), the entropy variables in terms of the primitive variables are:

$$\mathbf{V} = \frac{p^*}{p}\begin{pmatrix} U_3 + \frac{\alpha-1}{\gamma-1}p \\ -U_2 \\ U_1 \end{pmatrix} \equiv \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} \tag{182}$$

where

$$p^* = \frac{\gamma-1}{\alpha}\left(V_1 - \frac{1}{2}\frac{V_2^2}{V_3}\right) \tag{183}$$

The inverse transformation is given by:

$$\mathbf{U} = \frac{p}{p^*}\begin{pmatrix} V_3 \\ -V_2 \\ V_1 - \frac{\alpha-1}{\gamma-1}p^* \end{pmatrix} \tag{184}$$

Assuming a calorically-perfect gas,

$$E = \rho\left(c_v T - \frac{1}{2}u^2\right) \tag{185}$$

where T is the temperature (denoted θ in [9]) . Let us work this out in terms of the variables $\mathbf{U}$ and $\mathbf{V}$:

$$\begin{array}{rcl} E & = & \rho c_v T + \frac{\rho u^2}{2} \\ U_3 & = & U_1 c_v T + \frac{U_2^2}{2U_1} \\ V_1 - \frac{\alpha-1}{\gamma-1}p^* & = & V_3 c_v T + \frac{1}{2}\frac{V_2^2}{V_3} \\ V_1 - \frac{\alpha-1}{\gamma-1}\frac{\gamma-1}{\alpha}\left(V_1 - \frac{1}{2}\frac{V_2^2}{V_3}\right) & = & V_3 c_v T + \frac{1}{2}\frac{V_2^2}{V_3} \\ \left(V_1 - \frac{1}{2}\frac{V_2^2}{V_3}\right) - \frac{\alpha-1}{\alpha}\left(V_1 - \frac{1}{2}\frac{V_2^2}{V_3}\right) & = & V_3 c_v T \\ \frac{1}{\alpha}\left(V_1 - \frac{1}{2}\frac{V_2^2}{V_3}\right) & = & V_3 c_v T \end{array} \tag{186}$$

Rearranging,

$$c_v T = \frac{1}{\alpha}\left(\frac{V_1}{V_3} - \frac{1}{2}\frac{V_2^2}{V_3^2}\right) \tag{187}$$

Now, the heat flux is given by: $q_i = -\kappa T_{,i}$. Therefore, we must differentiate T with respect to x_i:

$$\begin{aligned} c_v \frac{\partial T}{\partial x_i} &= \frac{1}{\alpha}\left(\frac{V_3 V_{1,i} - V_1 V_{3,i}}{V_3^2} - \frac{V_3^2 V_2 V_{2,i} - V_2^2 V_3 V_{3,i}}{V_3^4}\right) \\ &= \frac{1}{\alpha}\left(\frac{1}{V_3} V_{1,i} - \frac{V_2}{V_3} V_{2,i} + \left(\frac{V_3 - V_1}{V_3^2}\right) V_{3,i}\right) \\ &= \frac{1}{\alpha}\begin{pmatrix} \frac{1}{V_3} & -\frac{V_2}{V_3} & \frac{V_3 - V_1}{V_3^2} \end{pmatrix} \mathbf{V}_{,i} \end{aligned} \tag{188}$$

so that, in the notation of [9], the heat flux matrix is:

$$\tilde{\mathbf{K}}_{11}^h = -\frac{\kappa}{\alpha c_v}\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{V_3} & -\frac{V_2}{V_3} & \frac{V_3 - V_1}{V_3^2} \end{pmatrix} \tag{189}$$

The matrix (189) is asymmetric, as Hughes asserts; but that in itself is not a problem, since the stability proof (Theorem 3.3.1) simply requires $\tilde{\mathbf{K}}_{11}^h$ to be positive semi-definite. Note that, for any $\mathbf{x} \in \mathbb{R}^3$:

$$\mathbf{x}^T \tilde{\mathbf{K}}_{11} \mathbf{x} = \mathbf{x}^T \left(\underbrace{\frac{\tilde{\mathbf{K}}_{11}^h + (\tilde{\mathbf{K}}_{11}^h)^T}{2}}_{\equiv (\mathbf{K}_{11}^h)^{symm}} \right) \mathbf{x} \tag{190}$$

Therefore, $\mathbf{x}^T \tilde{\mathbf{K}}_{11} \mathbf{x} \geq 0$ if the symmetric part of $\tilde{\mathbf{K}}_{11}^h$ is positive semidefinite. From (189),

$$\left(\tilde{\mathbf{K}}_{11}^h\right)^{symm} = -\frac{\kappa}{\alpha c_v}\begin{pmatrix} 0 & 0 & \frac{1}{2V_3} \\ 0 & 0 & -\frac{V_2}{2V_3} \\ \frac{1}{2V_3} & -\frac{V_2}{2V_3} & \frac{V_3 - V_1}{V_3^2} \end{pmatrix} \tag{191}$$

The eigenvalues of this matrix are:

$$\{\lambda_1, \lambda_2, \lambda_3\} = \left\{ 0, \frac{\kappa}{\alpha c_v}\left(\frac{V_1 - V_3 \pm \sqrt{V_1^2 - 2V_1 V_3 + 2V_3^2 + V_3^2 V_2^2}}{2V_3^2} \right) \right\} \tag{192}$$

The first eigenvalue is 0, so we are good to go with that one. Let us look at λ_1. It is non-negative is $V_1 - V_3 \geq 0$.

$$\begin{aligned} V_1 - V_3 &= \frac{p^*}{p}\left(U_3 + \tfrac{\alpha - 1}{\gamma - 1} p - U_1\right) \\ &= \frac{p^*}{p}\left(E + \tfrac{\alpha - 1}{\gamma - 1} p - \rho\right) \\ &= \frac{p^*}{p}\left(\tfrac{1}{\gamma - 1} p + \tfrac{1}{2}\rho u^2 + \tfrac{\alpha - 1}{\gamma - 1} p - \rho\right) \\ &= \frac{p^*}{p}\left(\tfrac{1}{2}\rho u^2 + \tfrac{\alpha}{\gamma - 1} p - \rho\right) \end{aligned} \tag{193}$$

Now, we desire $-\rho + \frac{\alpha}{\gamma - 1} p + \frac{1}{2}\rho u^2 \geq 0$. Let us see what requirements on α this constraint places:

$$\begin{aligned} -\rho + \tfrac{\alpha}{\gamma - 1} p + \tfrac{1}{2}\rho u^2 &\geq 0 \\ \tfrac{\alpha}{\gamma - 1} p &\geq \rho\left(1 - \tfrac{1}{2} u^2\right) \\ &\geq -\rho \tfrac{1}{2} u^2 \\ \alpha &\geq -\frac{\rho}{p} \tfrac{1}{2} u^2 (\gamma - 1) \end{aligned} \tag{194}$$

The right-hand side of (194) is necessarily negative given the physics ($p, \rho > 0$, $\gamma > 1$); therefore if $\alpha > 0$, then $\lambda_2 \geq 0$. Harten has already placed this constraint on α so in fact it is nothing new. Therefore $\lambda_2 \geq 0$ for $\alpha > 0$.

Let us examine the last eigenvalue λ_3. For it to be non-negative, we must have:

$$\begin{aligned} V_1 - V_3 &\geq \sqrt{V_1^2 - 2V_1V_3 + 2V_3^2 + V_3^2V_2^2} \\ V_1^2 - 2V_1V_3 + V_3^2 &\geq V_1^2 - 2V_1V_3 + 2V_3^2 + V_3^2V_2^2 \\ 0 &\geq V_3^2 + V_3^2V_2^2 \\ 0 &\geq V_3^2(1 + V_2^2) \end{aligned} \tag{195}$$

The only way for (195) to hold is if $V_3 = 0$. Let us see if this is possible:

$$V_3 = \frac{p^*}{p}U_1 = \frac{p^*}{p}\rho \tag{196}$$

But requiring this to be zero would amount to requiring $\rho = 0$, which is non-physical. Therefore the last eigenvalue λ_3 will necessarily be negative, unfortunately. The heat flux matrix $\tilde{\mathbf{K}}_{11}$ is *not* positive semi-definite with the choice of Harten's homogeneous generalized entropy flux function (179).

6 Distribution

This Internal Memorandum is to be distributed to the following recipients:

Recipient	Mail Stop
Matthew F. Barone	1124
Matthew R. Brake	0346
Lawrence J. Dechant	0825
Basil Hassan	0382
Jeffrey L. Payne	0825
Daniel J. Segalman	0557
Heidi K. Thornquist	0316

References

[1] S. Adhikari, On Symmetrizable Systems of Second Kind, *J. Appl. Mech.*, **67**(4) (2000) 797–802.

[2] M.F. Barone, D.J. Segalman, H.K. Thornquist, Galerkin Reduced Order Models for Compressible Flow, *Abstract for the 46th AIAA Aerospace Sciences Meeting and Exhibit Preferred Session: Grouped with Other Papers on Reduced Order Models*, Sandia National Laboratories, Albuquerque, NM (2007).

[3] M.F. Barone, J.L. Payne, Methods for Simulation-based Analysis of Fluid-Structure Interaction, *SAND2005-6573*, Sandia National Laboratories, Albuquerque, NM (2005).

[4] M.F. Barone, I. Kalashnikova, D.J. Segalman, H.K. Thornquist, Galerkin Reduced Order Models for Compressible Flow with Structural Interaction, *AIAA 46th Aerospace Science Meeting and Exhibit*, AIAA 2008-0612, Reno, NV (2008).

[5] M. Gerritsen, P. Olsson. Designing an Efficient Solution Strategy for Fluid Flows: 1. A Stable High Order Finite Difference Scheme and Sharp Shock Resolution for the Euler Equations. *J. Comput. Phys.* **129**:245–262 (1996).

[6] A. Harten. On the symmetric form of systems of conservations laws with entropy. *J. Comput. Phys.* **49**: 151–164 (1983).

[7] R.F. Heinemann, A.B. Poore. Multiplicity, Stability, and Oscillatory Dynamics of the Tubular Reactor. *Chem. Engng. Sci.* **36**:1411–1419 (1981).

[8] P. Holmes, J.L. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, New York, NY: Cambridge University Press, 1996.

[9] T.J.R. Hughes, L.P. Franca, M. Mallet. A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Comput. Meth. Appl. Mech. Engng.* **54**:223–234 (1986).

[10] I. Kalashnikova, Galerkin Reduced Order Model (ROM) for Compressible Flow: Enforcement of Boundary Conditions and Analysis of Coupled Fluid/Structure System, *SAND 5255161*, Sandia National Laboratories, Albuquerque, NM (2007).

[11] I. Kalashnikova. A Mathematical Analysis of the Well-Posedness, Stability and Convergence of a Galerkin Reduced Order Model (ROM) for Coupled Fluid/Structure Interaction Problems, *SAND 2008-5799P*, Sandia National Laboratories, Albuquerque, NM (2008).

[12] I. Kalashnikova, M.F. Barone. On the Stability and Convergence of a Galerkin Reduced Order Model (ROM) of Compressible Flow with Solid Wall and Far-Field Boundary Treatment. *Int. J. Numer. Meth. Engng.* (submitted April 2009).

[13] K. Kunisch, S. Volkwein, Galerkin Proper Orthogonal Decomposition for a General Equation in Fluid Dynamics, *SIAM J. Numer. Anal.*, **40**(2) 492–515 (2002).

[14] J.L. Lumley, Stochastic Tools in Turbulence. New York, NY: Academic Press, 1971.

[15] N.C. Nguyen, A.T. Patera, J. Peraire. A 'best points' interpolation method for efficient approximation of parametrized functions. *Int. J. Numer. Meth. Engng.* **73**:521–543 (2008).

[16] N.C. Nguyen, J. Peraire. An efficient reduced-order modeling approach for non-linear parametrized partial differential equations. *Int. J. Numer. Meth. Engng.* **76**:27–55 (2008).

[17] M. Rathinam, L.R. Petzold, L.R. A New Look at Proper Orthogonal Decomposition, *SIAM J. Numer. Anal.* **41** (5) 1893–1925 (2003).